NAS1.55:3053

0830-H-10
319

NASA Conference Publication 3053

MAY 31 1990

COMPLETED

Visual Information Processing for Television and Telerobotics

*Proceedings of a workshop held in
Williamsburg, Virginia
May 10-12, 1989*

NASA

NASA Conference Publication 3053

Visual Information Processing for Television and Telerobotics

Edited by
Friedrich O. Huck
NASA Langley Research Center
Hampton, Virginia

Stephen K. Park
College of William and Mary
Williamsburg, Virginia

Proceedings of a workshop sponsored by the
National Aeronautics and Space Administration,
Washington, D.C., and held in
Williamsburg, Virginia
May 10–12, 1989

National Aeronautics and
Space Administration
Office of Management
Scientific and Technical
Information Division

1989

I

FOREWORD

It is important in the conduct of research and development, as in many other human endeavors, to look around and review past and current activities that are related to one's own. As Lincoln once aptly put it: If we could first know where we are, and whither we are tending, we could then better judge what to do, and how to do it. Thus, the main objective of the 3-day workshop was to provide an informal forum in which researchers could be brought together to freely review and discuss the state of knowledge in various fields related to visual information processing. The emphasis was on applications to high-resolution television and vision-based telerobotics.

Visual information processing, as we understand it, begins when the radiance reflected or emitted by a scene strikes the optics of an image-gathering system. And it ends either (1) when the visual information has been recorded in the nervous system of human beings (e.g., via television) or (2) when some spatial features have been extracted by computer processing for higher-level decisions (e.g., for robotics). The intervening chain of processing steps may be divided, for the convenience of analysis, into several stages, including image gathering, coding, transmission, reconstruction or restoration, display, and interpretation. However, if these stages were to be optimized as separate tasks, independent of each other, then the end-to-end system performance could be impaired unwittingly despite the care and ingenuity bestowed upon the optimization of each stage.

Research in visual information processing started in the early 1930's for telephotography and television. This research was concerned mostly with the end-to-end performance of image gathering and reconstruction, accounting meticulously for the trade-off between aliasing, blurring, and noise in image gathering and the trade-off between blurring and raster effects in image displays.

Digital image processing began in the early 1960's soon after the advent of computers. Machine vision began soon thereafter in the 1970's. Both image processing and machine vision will grow in importance as the computational capacity of computers increases and the goals of these disciplines become more ambitious.

Many applications concerned with television and telerobotics have become severely constrained by the bandwidth of communication channels and the capacity of storage devices. As a direct result, the need for data compression has continuously increased during the last 30 years. In retrospect, the "first-generation" statistical coding schemes inherited from signal processing have been limited to compressions typically ranging from 2 to 10, whereas the "second-generation" perceptual coding schemes influenced by models of the human visual system have shown compressions up to 100 or so - but only at the risk of significant losses in fidelity and visual quality. Furthermore, most second-generation coding schemes based on visual perception models require extensive processing, some perhaps hours on a supercomputer per image frame. In the future, it can be expected that statistical and perceptual coding schemes will merge to achieve realistic compromises between desired performance and required processing, and between imaginative concepts and practical implementations.

Whereas the early studies of telephotography and television emphasized end-to-end performance of image gathering and display, more recent studies emphasize specific tasks, e.g., image coding, image restoration, and feature extraction. This approach has led to a number of simplifying assumptions about input and output devices that seldom hold in practice. A good example is digital image restoration. The design of image-gathering systems allows for significantly insufficient sampling to avoid extensive blurring of the reconstructed image, and the design of image displays requires the trade-off between blurring and raster effects. Yet the image-restoration algorithms given in the prevalent digital image processing literature typically do not account for either of these constraints. Nevertheless, if these constraints are correctly accounted for in image restoration, then (1) the fidelity, resolution, contrast, and clarity of the restored image improves significantly, and (2) the preferred trade-off between aliasing, blurring, and noise in image gathering changes towards less aliasing at the cost of blurring. Hence, the performance of television and telerobotics often may be improved not only by optimizing various processing steps independent of each other but also by increasing the awareness of end-to-end system optimization.

Friedrich O. Huck
Stephen K. Park

CONTENTS

FOREWORD iii

SESSION I: IMAGE GATHERING, CODING, AND PROCESSING
Chairman: Stephen K. Park
College of William and Mary

LINEAR DIGITAL IMAGING SYSTEM FIDELITY ANALYSIS 3
Steve Park

MODEL-BASED QUANTIFICATION OF IMAGE QUALITY 11
Rajeeb Hazra, Keith W. Miller, and Stephen K. Park

OPTIMAL FOCAL-PLANE RESTORATION 23
Stephen E. Reichenbach and Stephen K. Park

CODED-APERTURE IMAGING IN NUCLEAR MEDICINE 33
Warren E. Smith, Harrison H. Barrett, and John N. Aarsvold

CONTEXT DEPENDENT ANTI-ALIASING IMAGE RECONSTRUCTION 47
P. R. Beaudet, A. Hunt, and N. Arlia

CONTEXT DEPENDENT PREDICTION AND CATEGORY ENCODING FOR DPCM IMAGE COMPRESSION 57
Paul R. Beaudet

IMAGE GATHERING AND CODING FOR DIGITAL RESTORATION: INFORMATION EFFICIENCY AND VISUAL QUALITY 71
Friedrich O. Huck, Sarah John, Judith A. McCormick,
and Ramkumar Narayanswamy

DATA COMPRESSION FOR THE MICROGRAVITY EXPERIMENTS 93
Khalid Sayood, Wayne A. Whyte, Jr., Karen S. Anderson,
Mary Jo Shalkhauser, and Anne M. Summers

DETECTION OF EDGES USING LOCAL GEOMETRY 109
J. A. Gualtieri and M. Manohar

SESSION II: VISION MODELS AND IMAGE CODING
Chairman: Andrew B. Watson
NASA Ames Research Center

HIGH COMPRESSION IMAGE AND IMAGE SEQUENCE CODING 121
Murat Kunt

IMAGE PROCESSING BY INTENSITY-DEPENDENT SPREAD (IDS) 133
Tom N. Cornsweet

FROM PRIMAL SKETCHES TO THE RECOVERY OF INTENSITY AND REFLECTANCE REPRESENTATIONS 145
Rachel Alter-Gartenberg, Ramkumar Narayanswamy, and Karen S. Nolker

APPLICATIONS OF THE IDS MODEL 165
Eleanor Kurrasch

ISOLATING CONTOUR INFORMATION FROM ARBITRARY IMAGES 177
Daniel J. Jobson

SESSION III: ADVANCED CONCEPTS, SYSTEMS, AND TECHNOLOGIES
Chairman: Paul D. Try
Science and Technology Corporation

PARALLEL ASYNCHRONOUS SYSTEMS AND IMAGE PROCESSING ALGORITHMS 191
D. D. Coon and A. G. U. Perera

NEURAL NETWORKS FOR DATA COMPRESSION AND INVARIANT IMAGE RECOGNITION 203
Sheldon Gardner

KNOWLEDGE-BASED IMAGING-SENSOR FUSION SYSTEM 215
George Westrom

A KNOWLEDGE-BASED MACHINE VISION SYSTEM FOR SPACE STATION AUTOMATION 231
Laure J. Chipman and H. S. Ranganath

A PROGRAMMABLE IMAGE COMPRESSION SYSTEM 241
Paul M. Farrelle

HYBRID LZW COMPRESSION 251
H. Garton Lewis, Jr. and William B. Forsyth

CONNECTIONIST MODEL-BASED STEREO VISION FOR TELEROBOTICS 261
William Hoff and Donald Mathis

SESSION I: IMAGE GATHERING, CODING AND PROCESSING
Chairman: *Stephen K. Park*, College of William and Mary

SESSION II: VISION MODELS AND IMAGE CODING

Chairman: *Andrew B. Watson*, NASA Ames Research Center

SESSION III: ADVANCED CONCEPTS, SYSTEMS AND TECHNOLOGIES
Chairman: *Paul D. Try*, Science and Technology Corporation

Linear Digital Imaging System Fidelity Analysis

Steve Park
College of William & Mary
Department of Computer Science
May 30, 1989

In this paper, the combined effects of image gathering, sampling, and reconstruction are analyzed in terms of image fidelity. The analysis is based upon a standard end-to-end linear system model which is sufficiently general so that the results apply to most line-scan and sensor-array imaging systems. Shift-variant sampling effects are accounted for with an expected value analysis based upon the use of a fixed deterministic input scene which is randomly shifted (mathematically) relative to the sampling grid. This random sample-scene phase approach has been used successfully by the author and associates in several previous related papers [1]–[4].

Formulation

The end-to-end linear model upon which the results of this paper are based is characterized by three independent system components, an input scene $f(x,y)$, an image gathering point spread function $h(x,y)$, and an image reconstruction point spread function $r(x,y)$. All three of these components are referenced to a common orthogonal spatial coordinate system (x,y) normalized so that the sampling interval in both directions is unity. That is, sampling occurs at the integer coordinates (m,n). Because of this normalizing convention, when the model is analyzed in the Fourier domain, the associated spatial frequencies (μ,ν) have units of cycles/pixel and the Nyquist (folding) frequency is 0.5.

For notational convenience two other components are introduced, the pre-sampling image $g(x,y)$, and the reconstructed image $f'(x,y)$. The end-to-end model that relates the input scene f to the output reconstructed image f' is then

$$f(x,y) \xrightarrow{*h} g(x,y) \xrightarrow{\text{sample}} g(m,n) \xrightarrow{*r} f'(x,y) \tag{1}$$

where the $*$ operator denotes 2-d spatial convolution and $g(m,n)$ is $g(x,y)$ sampling onto the pixel grid. This model is the basis for all the analysis that follows, and, consequently, the results of this paper are applicable to the fidelity analysis of any sampled imaging system whose performance is characterized by the equation

$$f'(x,y) = \big[\big[f(x,y) * h(x,y)\big]\text{comb}(x,y)\big] * r(x,y) \tag{2a}$$

where

$$\text{comb}(x,y) = \sum_m \sum_n \delta(x-m, y-n) \tag{2b}$$

is the conventional 2-d comb function which accounts for sampling.

Image Fidelity

A variety of metrics have been advocated to measure how well one image matches another. These metrics include the 1-norm

$$\|f-g\|_1 = \int_x \int_y |f(x,y) - g(x,y)|\, dx\, dy$$

the common RMS 2-norm

$$\|f-g\| = \left(\int_x \int_y |f(x,y) - g(x,y)|^2\, dx\, dy \right)^{1/2}$$

which generalizes for $p \neq 2$ to the p-norm

$$\|f-g\|_p = \left(\int_x \int_y |f(x,y) - g(x,y)|^p\, dx\, dy \right)^{1/p}$$

and which approaches the ∞-norm

$$\|f-q\|_\infty = \max_{x,y} |f(x,y) - g(x,y)|$$

in the limit as $p \to \infty$. Of these, the RMS norm $\|f-g\|$ is far and away the most common, presumably because it lends itself so well to mathematical analysis.

The RMS norm *squared*

$$\|f-g\|^2 = \int_x \int_y |f(x,y) - g(x,y)|^2\, dx\, dy \qquad (3)$$

is a measure of image fidelity [5]. Specifically, the conventional definition of fidelity is

$$\text{fidelity} = 1 - \frac{\|f-g\|^2}{\|f\|^2} \qquad (4)$$

The primary purpose of this paper is to illustrate how the method of sample-scene phase averaging can be used to derive expressions for the three fundamental "fidelity loss" metrics

$$\|f-y\|^2 \qquad \text{and} \qquad \|g-f'\|^2 \qquad \text{and} \qquad \|f-f'\|^2.$$

The first of these metrics is a measure of *image blur*, the common loss of high spatial frequencies caused, for example, by defocus [5]. The second is *sampling and reconstruction blur*, the loss of fidelity caused by sampling (aliasing) and imperfect reconstruction [1]. The third, and most important, metric is the *end-to-end blur*, the net loss of fidelity caused by the combined effects of image gathering, sampling and reconstruction [6], [7]. Each of these fidelity loss terms will be analyzed in order, beginning with image blur.

Image Blur

The conventional *continuous-continuous* model of image formation (image gathering) is that the process is both linear and shift-invariant. That is, f and g are related by a convolution as

$$g(x,y) = \int_{x'}\int_{y'} h(x-x', y-y') f(x',y') dx'\, dy' \tag{5a}$$

where $h(x,y)$ is the image gathering point spread function (PSF) conventionally normalized so that

$$\int_{x}\int_{y} h(x,y) dx\, dy = 1. \tag{5b}$$

This model is much more easily understood when expressed in the spatial frequency (μ,ν) domain as

$$\hat{g}(\mu,\nu) = \hat{h}(\mu,\nu)\hat{f}(\mu,\nu) \tag{6a}$$

where

$$\hat{g}(\nu,\mu) = \int_{x}\int_{y} g(x,y)\exp(-2\pi i(x\mu + y\nu)) dx\, dy \tag{6b}$$

is the Fourier transform of g and the transforms $\hat{h}$, $\hat{f}$ are defined analogously.

It is well known that the PSF h typically acts as a low-pass filter. As a result, g is a blurred copy of f and the extent of this image blur is

$$\|f-g\|^2 = \int_{x}\int_{y} |f(x,y) - g(x,y)|^2 dx\, dy \tag{7a}$$

which can be rewritten, using the energy (Parseval's) theorem, as

$$\|f-g\|^2 = \int_{\mu}\int_{\nu} |\hat{f}(\mu,\nu) - \hat{g}(\mu,\nu)|^2 d\mu\, d\nu. \tag{7b}$$

However, from equation (6a), this last equation can be written as

$$\|f-g\|^2 = \int_{\mu}\int_{\nu} |1 - \hat{h}(\mu,\nu)|^2 |\hat{f}(\mu,\nu)|^2 d\mu\, d\nu. \tag{7c}$$

Note that if some metric other than the $\|\cdot\|^2$ norm were used, the energy theorem would not be applicable and the corresponding easy transition from a spatial domain integral to a corresponding frequency domain integral would not be possible. As the following discussion illustrates, this easy transition is a powerful argument in favor of the squared RMS metric. That is, the insight provided by equation (7c) is profound.

- Both terms in the integral are non-negative. Therefore,

$$\|f-g\|^2 = 0 \iff |1-\hat{h}(\mu,\nu)|^2|\hat{f}(\mu,\nu)|^2 = 0 \qquad \text{for } \textit{all } (\mu,\nu).$$

- Image blur is significant $\iff$ the scene has significant energy $|\hat{f}(\mu,\nu)|^2$ at spatial frequencies (μ,ν) where the optical transfer function (OTF) $\hat{h}(\mu,\nu)$ is significantly different from 1.
- Although the scene energy tends to decrease rapidly with increasing spatial frequency, most "natural" scenes have energy at *all* spatial frequencies. That is, natural scenes are not band-limited.
- The OTF typically decreases smoothly in magnitude from 1 at low spatial frequencies to 0 at high frequencies. Thus image blur is caused by a suppression of moderate to high spatial frequencies.

All these observation are well-known. However, the point is that they follow immediately by inspection of the frequency domain integral equation for $\|f-g\|^2$. This observation is the motivation for a search to find analogous equations for $\|g-f'\|^2$ and $\|f-f'\|^2$.

Sampling

The conventional ***continuous-discrete-continuous*** (end-to-end) model of image gathering, sampling and reconstruction is the convolution equation

$$f'(x,y) = \sum_m \sum_n r(x-m, y-n) g(m,n) \tag{8a}$$

where f' is the (continuous) reconstructed image and (as before) $g = h * f$. The (discrete-to-continuous) reconstruction process is conventionally assumed to be both linear and shift-invariant. It is therefore completely characterized by the reconstruction point spread function r conventionally normalized so that

$$\int_x \int_y r(x,y) dy\, dy = 1. \tag{8b}$$

This PSF can be thought of as the (continuous) output corresponding to a (discrete) sampled input which is 1 at the origin ($m = n = 0$) of the sampling grid and 0 at all other grid points. The reconstruction function is a low-pass filter which accounts for the combined effects of all post-sampling operations such as resampling and display.

The (continuous-to-discrete) sampling process is linear. However,

sampling is **not** a shift-invariant process.

That is, sampling causes the end-to-end system to be shift-variant. This sample-scene phase dependence complicates the end-to-end analysis significantly. For example, the end-to-end fidelity loss expression that one would write by analogy with equation (7c) is

$$\|f-f'\|^2 \neq \int_\mu \int_\nu |1-\hat{h}(\mu,\nu)\hat{r}(\mu,\nu)|^2 |\hat{f}(\mu,\nu)|^2 d\mu\, d\nu.$$

However (except in special cases) this equation is *not* correct.

Although the end-to-end model is not shift-invariant, it can be demonstrated that by using sample-scene phase averaging the metrics $\|f - f'\|^2$ and $\|g - f'\|^2$ can be written as

$$\|f - f'\|^2 = \int_\mu \int_\nu [\text{non-negative}]\, |\hat{f}(\mu,\nu)|^2 \, d\mu\, d\nu \tag{9a}$$

and

$$\|g - f'\|^2 = \int_\mu \int_\nu [\text{non-negative}]\, |\hat{g}(\mu,\nu)|^2 \, d\mu\, d\nu. \tag{9b}$$

Sample-Scene Phase Averaging

As first established in references [1] and [2], sample-scene phase averaging consists of the following steps.

- Fix the sampling grid.
- Shift the scene a random amount (u, v) relative to the fixed sampling grid

  $$f(x,y) \to f(x-u, y-v).$$

- Calculate (in the frequency domain) the corresponding shifted pre-sampling image

  $$g(x,y) \to g(x-u, y-v)$$

  and reconstructed image

  $$f'(x,y) \to f'(x,y;\, u,v).$$

- Assume that the random u and v shifts are independently and uniformly distributed between 0 and 1.
- Calculate (in the frequency domain) the expected values

  $$E\left[\|f - f'\|^2\right] = \int_0^1 \int_0^1 \|f - f'\|^2 \, du\, dv$$

  and

  $$E\left[\|g - f'\|^2\right] = \int_0^1 \int_0^1 \|g - f'\|^2 \, du\, dv.$$

- Observe that the image blur is independent of the sample-scene phase so that

  $$E\left[\|f - g\|^2\right] = \|f - g\|^2.$$

The results of this process are expected value equations consistent with (9a) and (9b).

Sampling and Reconstruction Blur

By using sample-scene phase averaging, it can be shown that

$$E\left[\|g - f'\|^2\right] = \int_\mu \int_\nu \left[|1 - \hat{r}(\mu,\nu)|^2 + \sum_m \sum_n{}' |\hat{r}(\mu - m, \nu - n)|^2 \right] |\hat{g}(\mu,\nu)|^2 \, d\mu \, d\nu \qquad (10)$$

where the double summation is over all $(m,n) \neq (0,0)$. However, an algebraically equivalent representation provides more insight into the fidelity loss associated with sampling and reconstruction. That is

$$E\left[\|g - f'\|^2\right] = \epsilon_s^2 + \epsilon_r^2 \qquad (11a)$$

where

$$\epsilon_s^2 = \int_\mu \int_\nu \left[\sum_m \sum_n{}' |\hat{g}(\mu - m, \nu - n)|^2 \right] |\hat{r}(\mu,\nu)|^2 \, d\mu \, d\nu \qquad (11b)$$

and

$$\epsilon_r^2 = \int_\mu \int_\nu |1 - \hat{r}(\mu,\nu)|^2 |\hat{g}(\mu,\nu)|^2 \, d\mu \, d\nu. \qquad (11c)$$

These two terms can be interpreted as follows [1].

- The term ϵ_s^2 accounts for aliasing caused by undersampling; it measures the loss of fidelity caused by the folding of significant image energy $|\hat{g}(\mu,\nu)|^2$ beyond the Nyquist frequency into those (low) frequencies where the reconstruction filter response $\hat{r}(\mu,\nu)$ is not 0. Moreover

  $$\epsilon_s^2 = 0 \iff \left[\sum_m \sum_n{}' |\hat{g}(\mu - m, \nu - n)|^2 \right] |\hat{r}(\mu,\nu)|^2 = 0 \qquad \text{for all } (\mu,\nu).$$

- The term ϵ_r^2 accounts for imperfect reconstruction; it measures the loss of fidelity caused by the presence of significant image energy at those (high) frequencies where $\hat{r}(\mu,\nu)$ is not 1. Moreover

  $$\epsilon_r^2 = 0 \iff |1 - \hat{r}(\mu,\nu)|^2 |\hat{g}(\mu,\nu)|^2 = 0 \qquad \text{for all } (\mu,\nu).$$

- If it were possible to produce a truly band-limited and sufficiently sampled image g, and if the reconstruction function was then taken to be $r(x,y) = \text{sinc}(x)\text{sinc}(y)$ then these two terms would be 0. (This is the sampling theorem.)

End-To-End Blur

In a similar manner, by using sample-scene phase averaging it can be shown that $E\left[\|f - f'\|^2\right]$ is

$$\int_\mu \int_\nu [\text{non-negative}] |\hat{f}(\mu,\nu)|^2 \, d\mu \, d\nu \qquad (12a)$$

where the [non-negative] term is

$$\left[|1-\hat{h}(\mu,\nu)\hat{r}(\mu,\nu)|^2 + |\hat{h}(\mu,\nu)|^2 \sum_m {\sum_n}' |\hat{r}(\mu-m,\nu-n)|^2\right] \tag{12b}$$

and again the summation is over all $(m,n) \neq (0,0)$. Also, as before, an algebraically equivalent representation provides more insight into the end-to-end fidelity loss. That is

$$E\left[\|f-f'\|^2\right] = \epsilon_s^2 + \epsilon_e^2 \tag{13a}$$

where ϵ_s^2 is the sampling (aliasing) term defined previously and

$$\epsilon_e^2 = \int_\mu \int_\nu |1-\hat{h}(\mu,\nu)\hat{r}(\mu,\nu)|^2 |\hat{f}(\mu,\nu)|^2 \, d\mu \, d\nu. \tag{13b}$$

This new term can be interpreted as follows.

- It accounts for the end-to-end loss of fidelity caused by significant scene energy at (mid to high) frequencies where the cascaded response, $\hat{h}(\mu,\nu)\hat{r}(\mu,\nu)$ is not 1. Moreover,

$$\epsilon_e^2 = 0 \iff |1-\hat{h}(\mu,\nu)\hat{r}(\mu,\nu)|^2 |\hat{f}(\mu,\nu)|^2 = 0 \qquad \text{for all } (\mu,\nu).$$

- It measures how well the reconstruction filter $\hat{r}$ is able to "deblur" (restore) those spatial frequencies which were suppressed prior to sampling by the image gathering OTF $\hat{h}$.

There is an inevitable trade-off here. For a fixed scene f and sampling grid, any attempt to decrease ϵ_e^2 by modifying $\hat{h}$ and $\hat{r}$ will result in an increase in ϵ_s^2 and conversely.

Fidelity Loss Budget

All of the previous analysis can be summarized in a *fidelity loss budget* given by the three sample-scene phase averaged metrics

$$E\left[\|f-g\|^2\right] = \epsilon_i^2 \tag{14a}$$

$$E\left[\|g-f'\|^2\right] = \epsilon_s^2 + \epsilon_r^2 \tag{14b}$$

$$E\left[\|f-f'\|^2\right] = \epsilon_s^2 + \epsilon_e^2 \tag{14c}$$

where

$$\epsilon_i^2 = \int_\mu \int_\nu |1-\hat{h}(\mu,\nu)|^2 \, |\hat{f}(\mu,\nu)|^2 \, d\mu \, d\nu \tag{14d}$$

$$\epsilon_s^2 = \int_\mu \int_\nu |\hat{h}(\mu,\nu)|^2 \left[\sum_m {\sum_n}' |\hat{r}(\mu-m,\nu-n)|^2\right] |\hat{f}(\mu,\nu)|^2 \, d\mu \, d\nu \tag{14e}$$

$$\epsilon_r^2 = \int_\mu \int_\nu |\hat{h}(\mu,\nu)|^2 \, |1-\hat{r}(\mu,\nu)|^2 \, |\hat{f}(\mu,\nu)|^2 \, d\mu \, d\nu \tag{14f}$$

$$\epsilon_e^2 = \int_\mu \int_\nu |1-\hat{h}(\mu,\nu)\hat{r}(\mu,\nu)|^2 \, |\hat{f}(\mu,\nu)|^2 \, d\mu \, d\nu. \tag{14g}$$

The four ϵ^2 terms can be easily calculated via numerical integration. All that is required is a knowledge of the scene energy $|\hat{f}(\mu,\nu)|^2$, image gathering OTF $\hat{h}(\mu,\nu)$ and reconstruction filter $\hat{r}(\mu,\nu)$—and ready access to a computer with a fast CPU and sufficient memory.

The four ϵ^2 terms are all interrelated and any attempt to minimize one must be carefully weighted against the potential increase of the others. Trade-off studies like this are the stuff of digital imaging system design.

References

[1] S.K. Park and R.A. Schowengerdt
Image Sampling, Reconstruction, and the Effect of Sample-Scene Phasing
Applied Optics, 21, 3142–3151, 1982.

[2] S.K. Park and R.A. Schowengerdt
Image Reconstruction by Parametric Cubic Convolution
Computer Vision, Graphics and Image Processing, 23, 258–272, 1983.

[3] R.A. Schowengerdt, S.K. Park and R. Gray
Topics in the Two-dimensional Sampling and Reconstruction of Images
International Journal of Remote Sensing, 5, 2, 333–347, 1984.

[4] S.K. Park, R.A. Schowengerdt and M.A. Kaczynski
Modulation-Transfer-Function Analysis for Sampled Image Systems
Applied Optics, 23, 2571–2582, 1984.

[5] E.H. Linfoot
Quality Evaluations of Optical Systems
Optica Acta, 5, 1–14, 1958.

[6] F.O. Huck, C.L. Fales, N. Haylo, R.W. Samms and K. Stacy
Image Gathering and Processing: Information and Fidelity
Journal of the Optical Society of America, A2, 1644–1666, 1985.

[7] F.O. Huck, C.L. Fales, J.A. McCormick and S.K. Park
Image-Gathering System Design for Information and Fidelity
Journal of the Optical Society of America, A5, 285–299, 1988.

Bibliography

[A] P.B. Fellgett and E.H. Linfoot
On the Assessment of Optical Images
Philosophical Transactions of the Royal Society of London 247, 369–407, 1955.

[B] F.O. Huck and S.K. Park
Optical-Mechanical Line-Scan Imaging Process: Its Information Capacity and Efficiency
Applied Optics, 14, 2508–2520, 1975.

[C] F.O. Huck, N. Haylo and S.K. Park
Aliasing and Blurring in 2-D Sampled Imagery
Applied Optics, 19, 2174–2181, 1980.

Model-Based Quantification of Image Quality

Rajeeb Hazra, Keith W. Miller, Stephen K. Park

Introduction [1]

In 1982, Park and Schowengerdt [1] published an end-to-end analysis of a digital imaging system quantifying three principal degradation components (i) *image blur* - blurring caused by the acquisition system (ii) *aliasing* - caused by insufficient sampling and (iii) *reconstruction blur* - blurring caused by the imperfect interpolative reconstruction. This analysis, which measures degradation as the square of the radiometric error, includes the sample-scene phase as an explicit random parameter and characterizes the image degradation caused by imperfect acquisition and reconstruction together with the effects of undersampling and random sample-scene phases. In a recent paper Mitchell and Netravelli [3] displayed the visual effects of the above mentioned degradations and presented subjective analysis about their relative importance in determining image quality.

The primary aim of the research in this paper is to use the analysis of Park and Schowengerdt [1],[8] to correlate their *mathematical* criteria for measuring image degradations with subjective *visual* criteria. Insight gained from this research can be exploited in the end-to-end design of optical systems, so that system parameters (transfer functions of the acquisition and display systems) can be designed relative to each other, to obtain the "best possible" results using quantitative measurements.

Formulation

In this section we present an end-to-end model of a digital imaging system. This model was used by Park and Schowengerdt [1] to derive expressions for the degradation caused by the various components of the system.

The model upon which the results of this paper are based is described in Fig 1. The parameters u and v are explicit sample scene phase parameters which have the range of $\pm\frac{1}{2}$ for pixels placed at unit distance from each other. The action of the imaging subsystem is described by the convolution (denoted by $*$) of the system point spread function (PSF) $h(x,y)$ with the scene

$$g(x-u,y-v) = h(x,y) * f(x-u,y-v) \tag{1}$$

The image is then sampled onto a cartesian grid. This sampling operation is represented symbolically as the multiplication of the image with the *comb* or *Shah* function

$$g_s(x,y;u,v) = \sum_m \sum_n g(x-u,y-v)\delta(x-m,y-n) \tag{2}$$

The notation $g_s(x,y;u,v)$ expresses the fact that the sampling subsystem is not shift-invariant.

---

[1] This paper refers to research described in references 1-8.

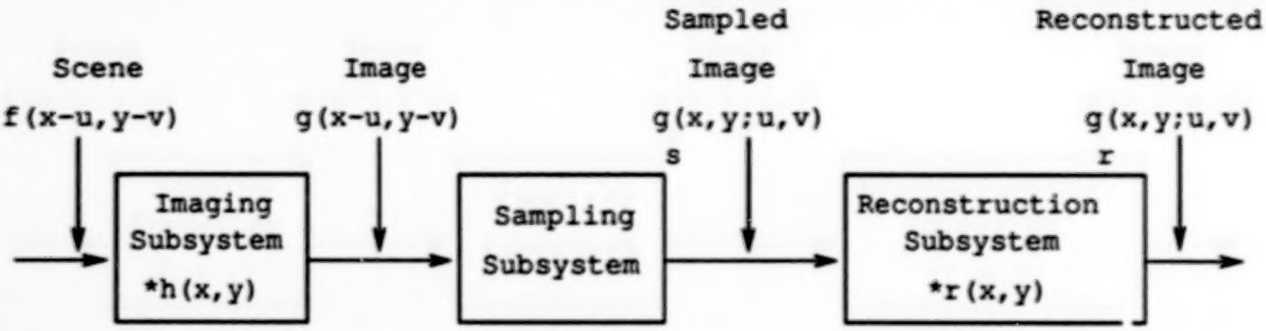


Figure 1: An Imaging, Sampling and Reconstruction System

We take the point of view that the reconstruction filter $r(x,y)$ is designed so that the reconstructed image is an accurate reproduction of the output of the imaging system. The reconstructed image is compared to the image g and not the scene f; thus, the reconstruction filter typically does not attempt to perform any restoration.

Reconstruction is also symbolically modeled as a convolution operation

$$g_r(x,y;u,v) \;=\; r(x,y) \;*\; g_s(x,y;u,v) \tag{3}$$

Park and Schowengerdt measure the *accuracy* of reconstruction as the mean square radiometric error and define the term

$$\epsilon_{SR}^2(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}[g(x-u,y-v) - g_r(x,y;u,v)]^2 dx dy \tag{4}$$

and analogously

$$\epsilon_I^2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}[f(x-u,y-v) - g(x-u,y-u)]^2 dx dy \tag{5}$$

where ϵ_{SR}^2 and ϵ_I^2 measure the *sampling-reconstruction* degradation and *image blur* respectively. As suggested by the notation, image blur is independent of the sample scene phase due to the shift invariance of the convolution operation. The sampling-reconstruction degradation is not.

Fourier analysis yields equivalent expressions for ϵ_{SR}^2 and ϵ_I^2 in the frequency domain. Park et al. showed that

$$\epsilon_I^2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mid 1 - \hat{h}(\nu_x,\nu_y) \mid^2 \mid \hat{f}(\nu_x,\nu_y) \mid^2 d\nu_x d\nu_y \tag{6}$$

where (ν_x,ν_y) are spatial frequencies (units of cycles per sampling interval), $\hat{h}(\nu_x,\nu_y)$ is the imaging subsystem OTF (optical transfer function) and $\mid \hat{f}(\nu_x,\nu_y) \mid$ is the magnitude of the transform of the scene.

The corresponding expression for the sampling and reconstruction degradation is given in terms of an ensemble of scenes formed by varying the sample scene phase parameters uniformly over their entire range. Thus, we obtain the *expected value* of this degradation in the form

$$E[\epsilon_{SR}^2] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^2(\nu_x,\nu_y) \mid \hat{h}(\nu_x,\nu_y) \mid^2 \mid \hat{f}(\nu_x,\nu_y) \mid^2 d\nu_x d\nu_y \tag{7}$$

where the term $e^2(\nu_x,\nu_y)$ accounts for the effects of imperfect reconstruction and undersampling and is given by

$$e^2(\nu_x,\nu_y) = \mid 1-\hat{r}(\nu_x,\nu_y) \mid^2 + \sum_{(m,n)\neq(0,0)}\sum \mid \hat{r}(\nu_x - m, \nu_y - n) \mid^2 \tag{8}$$

where $\hat{r}(\nu_x,\nu_y)$ is the reconstruction filter, i.e. the Fourier transform of $r(x,y)$. $E[\epsilon_{SR}^2]$ can be written as the sum of two terms,

$$E[\epsilon_{SR}^2] = \epsilon_R^2 + \epsilon_S^2 \tag{9}$$

where

$$\epsilon_R^2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mid 1-\hat{r}(\nu_x,\nu_y) \mid^2 \mid \hat{h}(\nu_x,\nu_y)\hat{f}(\nu_x,\nu_y) \mid^2 d\nu_x d\nu_y \tag{10}$$

and

$$\epsilon_S^2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \left[\sum_{m,n\neq 0,0}\sum \mid \hat{r}(\nu_x - m, \nu_y - n) \mid^2\right] \mid \hat{h}(\nu_x,\nu_y)\hat{f}(\nu_x,\nu_y) \mid^2 d\nu_x d\nu_y \tag{11}$$

The term ϵ_R^2 accounts for imperfect reconstruction while ϵ_S^2 accounts for aliasing due to undersampling.

Analysis and Visual Perception of Image and SR Blur

Image blur is caused by the non-ideal frequency response of the imaging subsystem. Eq. (6) is a mathematical statement of this fact. Almost invariably, the frequency response of an imaging system approaches zero at high frequencies and thus this subsystem acts as a low-pass filter. Image blur alone, uncoupled from sampling and reconstruction blur, is perceived as a loss of high frequency detail in the scene.

The average sampling and reconstruction blur, as suggested by Eq. (7) is caused by inadequacies in both the sampling and the reconstruction subsystem. The sampling contribution to this degradation is expressed by Eq. (11) which states that aliasing is caused by the presence of significant image energy at frequencies where the energy in the reconstruction filter sidebands

$$\sum_{m,n\neq 0,0}\sum \mid \hat{r}(\nu_x - m, \nu_y - n) \mid^2 \tag{12}$$

is not zero. This is illustrated in Fig 2 where the replicas of the image spectrum (formed by sampling) overlap and the reconstruction filter cannot isolate a pure version of the *base-band* spectrum. This type of degradation is sometimes called *prealiasing* [3] and will always be present if the image is not sufficiently sampled, even with perfect reconstruction.

Even when the replicated spectra do not overlap (i.e the image has been sufficiently sampled), image quality may suffer due to poor reconstruction, as illustrated in Fig 3. In this case, the response of the reconstruction filter is too broad and thus the reconstructed signal includes some (high) frequencies not present in the original image. This type of aliasing is sometimes called *postaliasing* [3]. When the image spectrum has significant power at frequencies very near the Nyquist (cutoff) frequency (i.e the image spectrum and its nearest replica come very close to each other), the design of the reconstruction filter becomes difficult as the roll-off has to be very sharp (resulting in a filter with a very large kernel in the spatial domain). This problem has been noted by several researchers [2], [3].

Pre- and postaliasing are often perceived as artifacts in the reconstructed scene [3]. However, it should be noted that in general, an absence of artifacts does not imply that there is no pre or postaliasing. Aliasing can manifest itself as blurring as well (due to attenuation of the high frequencies in the scene or image spectrum) and is almost impossible to differentiate from image blur.

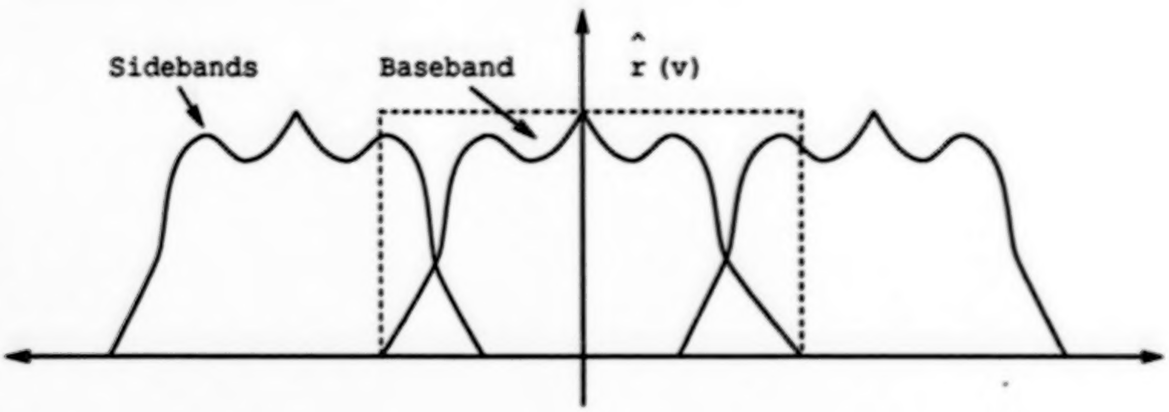

Sidebands
Baseband
$\hat{r}(v)$

Figure 2: Prealiasing resulting from insufficient sampling

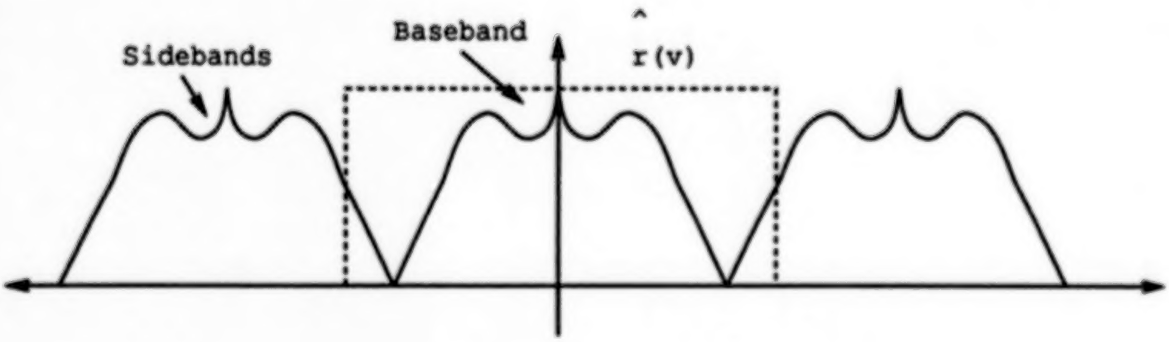

Sidebands
Baseband
$\hat{r}(v)$

Figure 3: Postaliasing resulting from poor reconstruction

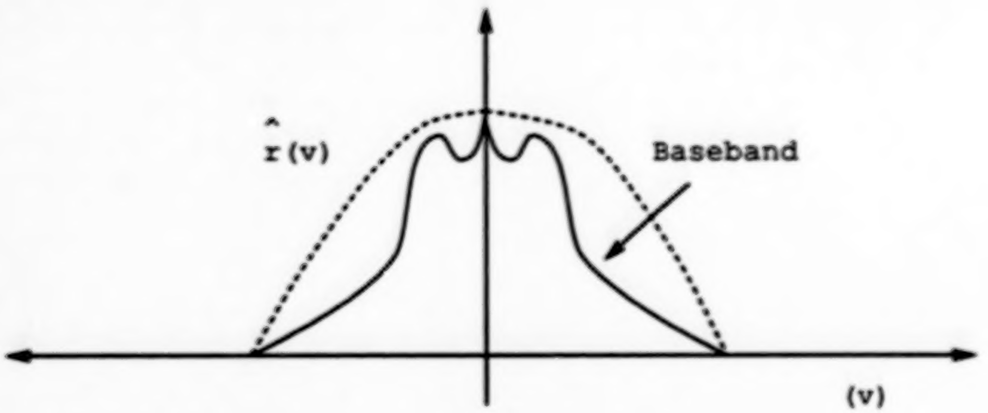


Figure 4: Baseband attenuation resulting from imperfect reconstruction

In addition to removing the sidebands of the signal spectrum, the reconstruction filter also needs to pass the original image spectrum base-band with minimal distortion (Fig 4). Eq. (10) states this idea formally. It measures the contribution to SR blur caused by the presence of significant image energy, $|\hat{h}(\nu_x,\nu_y)\hat{f}(\nu_x,\nu_y)|^2$ at frequencies where $\hat{r}(\nu_x,\nu_y) \neq 1$. This type of reconstruction error is known as ***base-band attenuation.*** This is analogous to image blur in the sense that the reconstruction filter acts as a low-pass filter resulting in a loss of high-frequency detail in the reconstructed image.

The problem of designing a good reconstruction filter is made difficult because an "ideal" filter is a sinc filter in the spatial domain. The sinc is quasi-ideal in the sense that a signal can be *perfectly* reconstructed from its samples by using sinc-interpolation only if the signal is bandlimited and sufficiently sampled. However, the sinc is impossible to realize in practice and finite approximations to it produce an effect commonly known as *ringing*. Ringing is perceived as rippling patterns radiating from high contrast edges [3] and is strongly suggested by the form of the impulse response of the sinc.

Another problem in designing a reconstruction filter is the problem of *sample-frequency ripple*. This problem can be best understood in terms of a uniformly gray image which is sampled and reconstructed to yield an image where the gray-level uniformity is destroyed. This is often perceived as spurious patterns on the background in an image. To eliminate this problem, it is necessary to design the reconstruction filter $\hat{r}(\nu_x,\nu_y)$ so that the equation

$$\sum_m \sum_n r(x-m, y-n) = 1 \tag{13}$$

is satisfied.

An important point to note in this discussion is that even though it is possible (at least in theory) to minimize image blur and sampling-reconstruction blur individually by suitable filter design, in an end-to-end system the subsystems cannot be designed in isolation from one another to minimize both image and sampling-reconstruction blur simultaneously. Eq. (6) suggests that image blur is minimized when $\hat{h}(\nu_x,\nu_y) = 1$ for all frequencies where there is non-zero scene energy. However, from Eq. (8) we see that the average sampling-reconstruction blur will be minimized when $\hat{h}(\nu_x,\nu_y) = 0$ at all those frequencies where the reconstruction filter does not have unit (perfect) response. These are conflicting requirements and a compromise has to be achieved based on the *relative visual importance* of the two types of degradation.

It has been observed [6], [7] that the response of human viewers to various spatial effects of filters is subjective. Filters that result in some aliasing and base-band attenuation have sometimes been observed to yield results which are visually pleasing to human viewers. There is evidence [6] that suggests that a moderate amount of ringing can "improve" the visual quality of an image by introducing an illusion of sharpness (high frequency), although in terms of the amount of degradation (which can be measured by Eq. 6 - 11), this may correspond to a higher mean squared error.

In our research, we attempt to correlate the mathematical criteria for optimal end-to-end processing with subjective visual testing for Gaussian transfer functions. There is evidence that people prefer some aliasing and ringing (which give an illusion of sharpness), but that people are sensitive to high-frequency suppression (blurring). The primary motivation of this study is to assess the effect of each of these individual degradation components on the quality of the reconstructed image. This research has application to the design of end-to-end imaging systems where the components can be tuned to obtain the best possible results. In the next section we describe our models for the various components of the system and simulation results.

Imaging and Reconstruction System Transfer Function Models

In order to simulate an end-to-end imaging system it is necessary to associate a model with the imaging (camera) subsystem and the reconstruction (display) subsystem;i.e,we need to assign a *functional* form to both $\hat{h}(\nu_x, \nu_y)$ and $\hat{r}(\nu_x, \nu_y)$. In the discussion that follows, we refer to $\hat{h}$ as the Camera Transfer Function (CTF) and $\hat{r}$ as the Display Transfer Function (DTF). In our analysis we have chosen to model the CTF and DTF as Gaussian functions of the form

$$e^{-\left[\frac{1}{r^2}(\nu_x^2+\nu_y^2)\right]} \tag{14}$$

where r is the parameter which controls the spread of the function. It can be shown that r is proportional to the standard deviation of the Gaussian function (σ_ν). They are related as

$$r = \sqrt{2}\sigma_\nu \tag{15}$$

The two-dimensional Gaussian function is separable and its Fourier transform is also a Gaussian.

In our model, the Gaussian is symmetric in the two dimensions resulting in the filter kernels being circularly symmetric. Thus, only a single parameter (r) is required to characterize each of the Gaussian functions representing the CTF and the DTF. Thus, due to the duality of the Gaussian and its Fourier transform, a broad frequency response can be achieved by a very small kernel in the spatial domain and vice versa.

The reason for modelling the CTF and the DTF as Gaussian functions is primarily due to the its popularity amongst designers of optical instruments [4], [5]. Several variations of the pure Gaussian (e.g. sharpened Gaussian and the sum of two Gaussians [4]) have been used as models for the transfer functions, especially for interpolative reconstruction systems. In our end-to-end model, we have three system parameters - the sampling rate and the standard deviations of the Gaussian camera and display transfer functions. These parameters can be varied to influence both image and sampling and reconstruction blur. End-to-end simulation using these models for the imaging and reconstruction subsystems thus allows us to study the interplay between the various degradations discussed in the previous section and correlate mathematical results (blur coefficients) with subjective (visual) judgements about image quality. The primary goal of this simulation study is to identify a relationship between the two parameters which will result in the best possible reconstructed image. Such a relationship can then be used as a design rule for end-to-end systems employing scanning and interpolative reconstruction.

Simulation Results

The numerical simulation of the imaging system described in Fig 1. has been performed on two images - one of cat's face and the other of the central portion of a dollar bill. The images are 512 x 512 pixels in dimension and are quantized to 16 bits/gray-level. For the purpose of display, these images have been rescaled into a gray-level range of 0 to 255 (8 bits/gray-level).

In the simulation of the imaging process, the Fourier spectra of these scenes have been multiplied with a Gaussian CTF to produce the corresponding image spectra. The reverse Fourier transform of the image spectra produces the corresponding image in the spatial domain. It is with this image that the final reconstructed scene is compared to judge the quality of reconstruction.

The sampling subsystem has been simulated by sub-sampling the 512 x 512 image down to 128 x 128. A uniform sampling scheme has been chosen primarily due to its simplicity as well as its popularity amongst designers of (digital) optical equipment.

Finally, the reconstruction process is implemented in a manner similar to the imaging process. The sampled images have been enlarged to 512 x 512 by zero-filling and their Fourier spectra have then been multiplied with a Gaussian DTF to produce the spectra of the reconstructed scenes. The inverse Fourier transform is then applied to these spectra to produce the reconstructed scenes in the spatial domain.

The parameter of interest for the transfer functions is r (Eq. 14.), which controls the standard deviation of the Gaussian functions used to model the CTF and the DTF. In order to study the degradation caused by these two subsytems, r_{CTF} and r_{DTF} have been varied over a range of values - the range selected is standardized with respect to the size of the sampled image (128 x 128).

The reconstructed scenes have been evaluated by about 20 people and the degradation values corresponding to their choice of the best possible reconstruction are shown on the corresponding plots. The observers were first shown the images after they were passed through the imaging subsystem and were then asked to find the most faithful reconstruction from the collection of processed images for different system parameters. The candidate images were displayed in a random order to eliminate any positional bias that may have been present. The contrasts of these images were also strictly matched to eliminate any contrast bias.

Fig 5 shows the cat image after being passed through the acquisition phase of an imaging system. Figs 6 - 8 show reconstructed images of this acquired image with different display subsystems. Figs 9 - 14 are plots of the different error components for the dollar and the cat images. These values have been calculated using Eq.(6)-(11) and the horizontal axis of the plots refer to a fraction of 128 which represents the value of r_{CTF}. Figs 15 - 17 show several processed versions of the dollar bill image. Fig 15 shows the original dollar bill which serves as the scene in our simulations. Fig 16 and Fig 17 show results from two opposite ends of the processing spectrum - Fig 16. shows the excessive blurring introduced by the narrow (frequency domain) transfer functions, while Fig 17 exhibits the characteristic sample-frequency ripple associated with a wide (frequency domain) display transfer function.

The preliminary results of the visual testing have yielded interesting "observations" about the visual impact of the different kinds of degradations that are inevitably introduced in a nonideal end-to-end imaging system; 18 out of the 20 observers chose the reconstructed image corresponding to $r_{CTF} = 76.8$ (i.e. *CTF FACTOR* = 0.6 in Fig 7) as the "best" reconstructed scene for the cat image. From the plot it is clear that these values of the parameters correspond to a situation where the total degradation is dominated by the sampling (or aliasing) blur. This reinforces the belief that the human eye is more critical of blurring (of any kind) than other types of degradations which introduce some high frequency features which are not present in the original image. None of the observers selected those images for which the image blur is the dominant degradation term. In particular, the sample-frequency ripple effect (which manifests itself as a fine wire mesh over the images) helped to a certain extent to create an illusion of feature (edge) sharpness that made the observers select images with a moderate amount of this effect as the "best" images.

The subjective evaluation suggests that to the untrained human eye, image blur (i.e. any supression of high frequencies) is often more annoying than sampling artifacts which may create an illusion of sharpness. However, much work still needs to be done. In planned extensions, we will control viewing conditions more stringently to eradicate some biases that may be reflected in our current results. We plan to use a digital monitor instead of film since we have experienced a great deal of contrast and texture variability with film. There is also the need for more exhaustive testing (more scenes with a greater variation in the frequency content etc.) under more controlled conditions. The end-to-end model must be improved to incorporate more sophisticated models of the acquisition and display subsystems as well as physcophysical parameters such as the *contrast sensitivity function*. The sophistication of the human subjects with respect to digital image processing fundamentals may also be a significant bias factor when testing certain images. Finally, the whole experiment would be incomplete unless an end-to-end (initial scene to the reconstructed scene) simulation is performed and the analytical results are correlated with visual testing.

Acknowledgments

The authors would like to thank several people who helped put this experiment together. In particular, they would like to express their gratitude to Steve Reichenbach for his help with the simulation software and Kathy Stacy, Des Leonard and Betsy Avis at NASA Langley Research Center for their help in processing the images. They also thank the graduate students in the Computer Science department of the College of William & Mary who helped with the subjective evaluations of the test images.

References

[1] Park S.K and Schowengerdt R.A, 1982 *Appl. Optics*,**21**,3142

[2] Cook R.L, *ACM Trans. Graphics*, Vol. 5 No. 1 Jan, 1986.

[3] Mitchell, D.P and Netravelli A, *ACM Trans. Graphics* Vol. 22 Aug, 1988.

[4] Schreiber W.F, *Fundamentals of Electronic Imaging Systems* Springer-Verlag, 1986.

[5] Harris F.C, *Proc. IEEE*, **66**, No. 1, Jan. 1978.

[6] Brown E. F, *J. of the SMPTE*, Vol 78, April 1969

[7] Schreiber W.F, Troxel D.E, *IEEE Trans. on PAMI*, PAMI-7, No. 2, March 1985

[8] Schowengerdt R.A, Park S.K and Gray R, 1984 *Int. J. Remote Sensing*, Vol 5. No. 2.

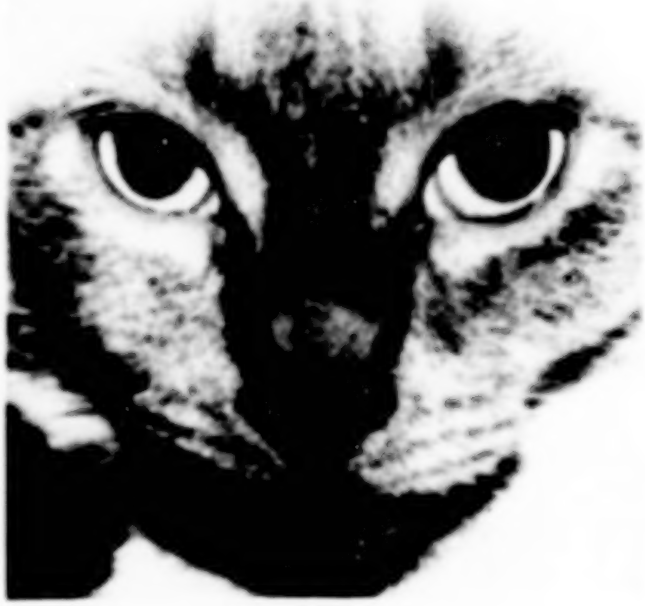


Fig. 5 : Original CAT image with

CTF FACTOR=0.6 and no SR degradation.

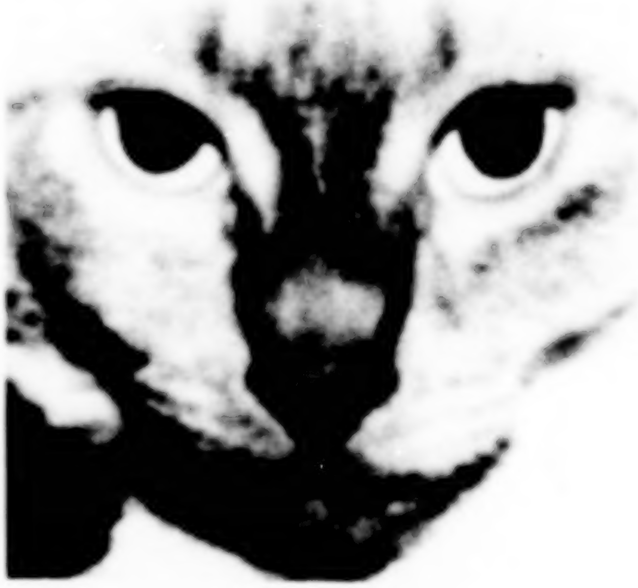


Fig 6 : Reconstructed scene with

CTF FACTOR=0.6 and DTF FACTOR=0.3

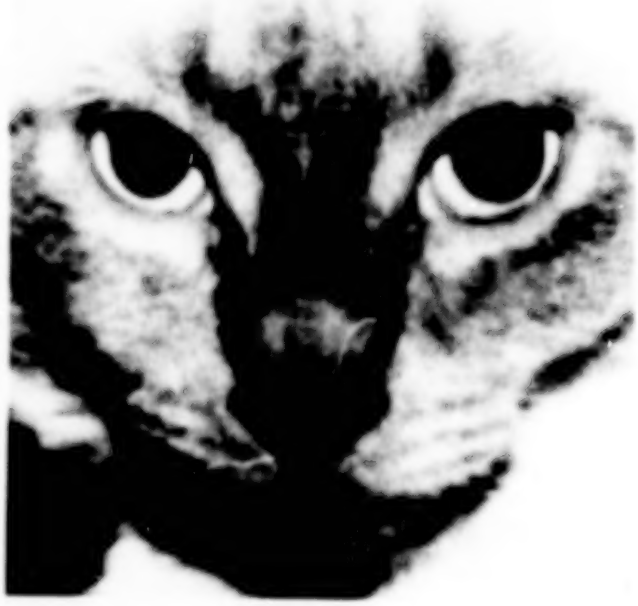


Fig 7 : Reconstructed scene with

CTF FACTOR = 0.6 and DTF FACTOR=0.6

(chosen as the best reconstruction of

Fig 5. by selected observers)

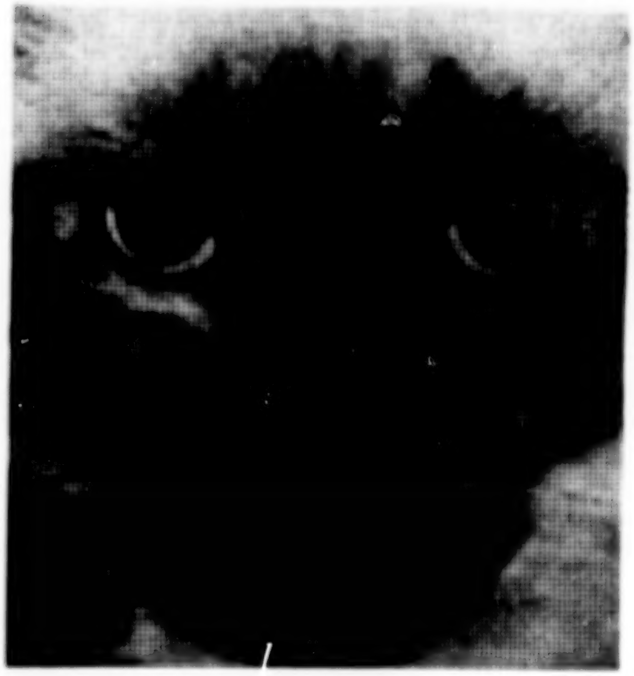


Fig 8 : Reconstructed scene with

CTF FACTOR=0.6 and DTF FACTOR=0.7

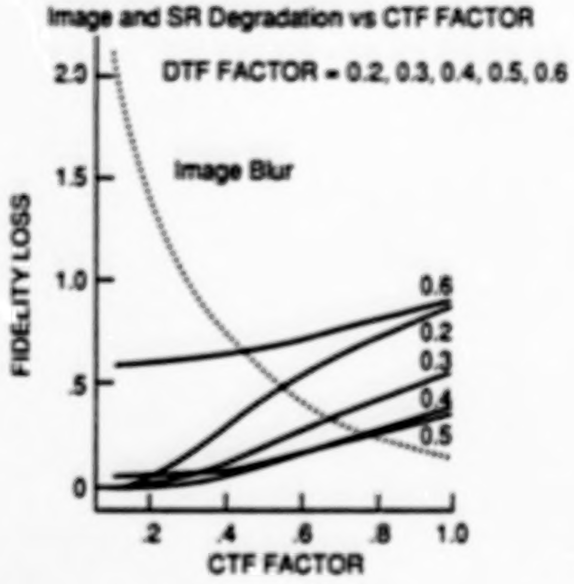


Figure 9: Image and SR Blur vs CTF FACTOR (Dollar)

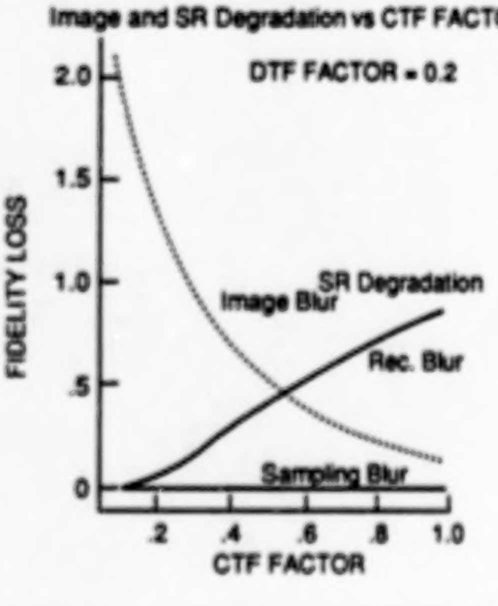


Figure 10: Image and SR Blur vs CTF FACTOR (Dollar)

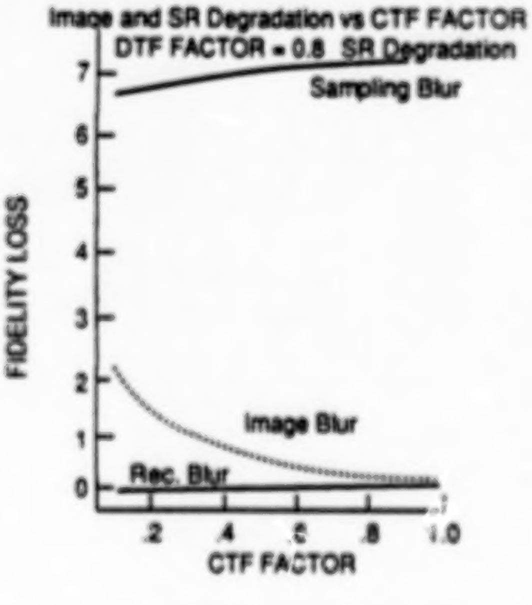


Figure 11: Image and SR Blur vs CTF FACTOR (Dollar)

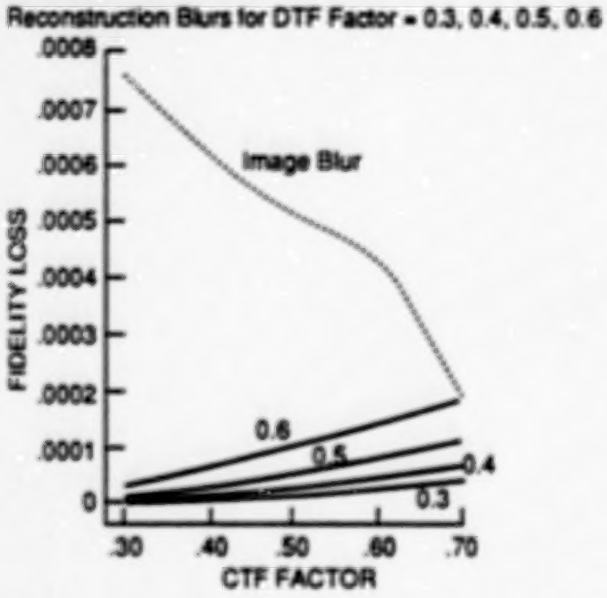


Figure 12: Reconstruction Blur vs CTF FACTOR (Cat)
(for DTF FACTOR = 0.3 to 0.6)

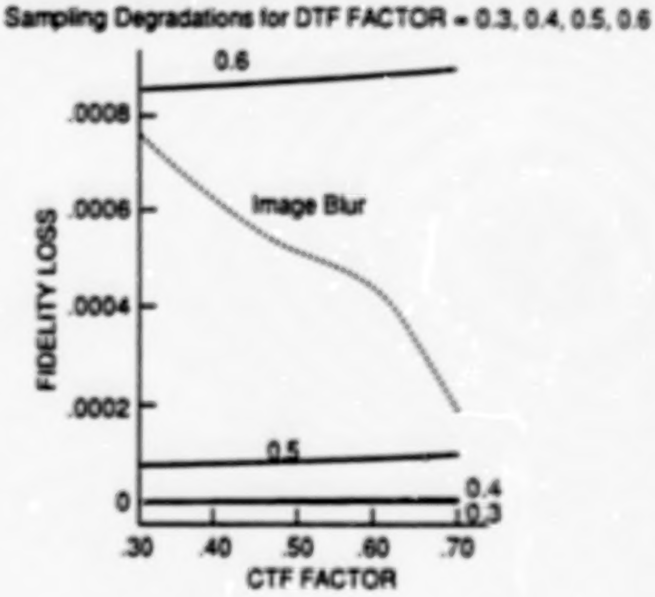


Figure 13: Sampling Blur vs CTF FACTOR (Cat)
(for DTF FACTOR = 0.3 to 0.6)

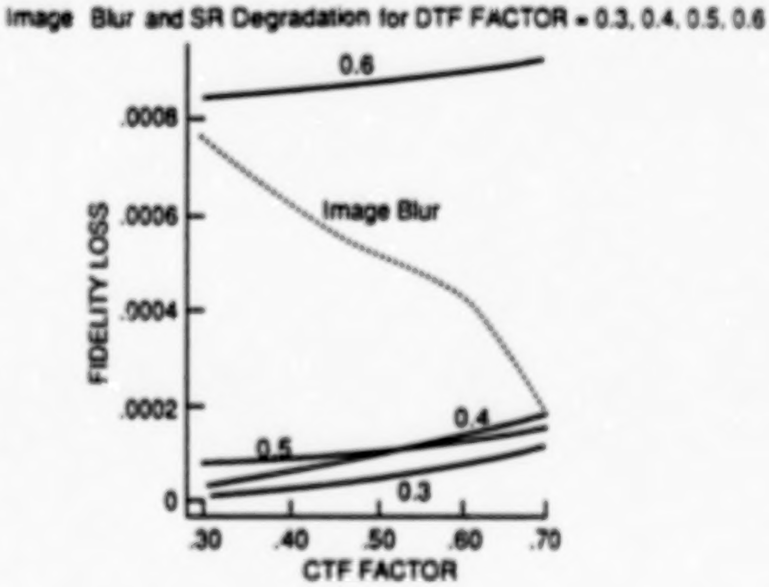


Figure 14: Image and SR Blur vs CTF FACTOR (Cat)
(for DTF FACTOR = 0.3 to 0.6)

Fig 15 : Original Dollar Image with no image and SR degradations.

Fig 16 : Reconstructed Scene with CTF FACTOR=0.3 and DTF FACTOR=0.3

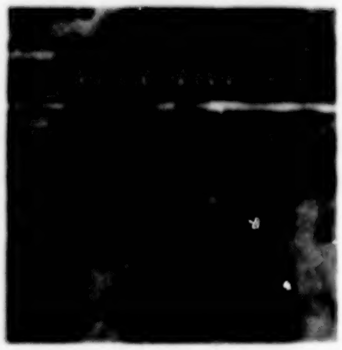


Fig 17 : Reconstructed Scene with CTF FACTOR=0.3 and DTF FACTOR=0.7

Optimal Focal-Plane Restoration

Stephen E. Reichenbach Stephen K. Park

Computer Science Department
College of William and Mary
Williamsburg, Virginia

Abstract

Image restoration can be implemented efficiently by calculating the convolution of the digital image and a small kernel during image acquisition. Processing the image in the focal-plane in this way requires less computation than traditional Fourier-transform-based techniques such as the Wiener filter and constrained least-squares filter. In this paper, the values of the convolution kernel that yield the restoration with minimum expected mean-square error are determined using a frequency analysis of the end-to-end imaging system. This development accounts for constraints on the size and shape of the spatial kernel and all the components of the imaging system. Simulation results indicate the technique is effective and efficient.

1 Introduction

The Wiener filter is probably the best known and most widely used restoration tool. Given a few assumptions and some knowledge of the system, the Wiener filter minimizes the expected mean-square-error (MSE) of the restoration. While MSE is by no means a perfect yardstick for restoration quality, it is a useful measure and leads to an optimal filter. In many applications, such as those requiring television-rate processing (30 images per second), the most serious drawback of the Wiener filter is its high computational cost. Small spatial kernels can be applied with much less computation. This paper describes the design of small restoration kernels that, within the spatial constraints, minimize restoration MSE.

2 End-to-End Analysis and Wiener Restoration

Traditionally, Wiener restoration has been based on a model of the imaging process with two components: the linear, shift-invariant point-spread function (PSF) of the image acquisition device and additive, signal-independent noise. This model ignores the significant impact of sampling and display reconstruction on image quality. A recent paper[1] presented a derivation of the Wiener filter that is based on a more accurate model of the end-to-end imaging process. This model is illustrated in Figure 1.

The end-to-end process is described equivalently by equations in either the spatial domain or frequency domain. The displayed (or resulting) image r is

$$r(x) = \frac{1}{N}\sum_{n'}\frac{1}{N}\sum_{n}\left(\frac{1}{N}\int_{-\infty}^{\infty} s(n-x')h(x')dx'\text{Ш}(n) + e[n]\right) f[n'-n]d(x-n') \tag{1}$$

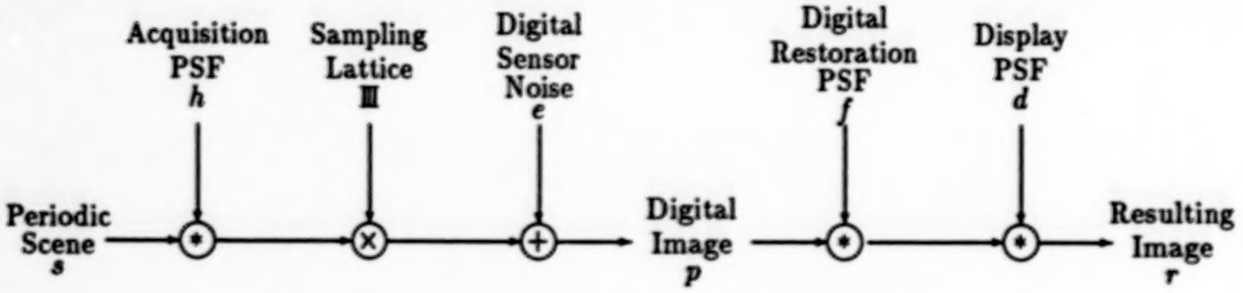

Acquisition PSF h
Sampling Lattice Ш
Digital Sensor Noise e
Digital Restoration PSF f
Display PSF d
Periodic Scene s
Digital Image p
Resulting Image r

Figure 1: End-to-End Imaging and Spatial Restoration Model

Assuming the scene s is periodic, the equivalent frequency domain expression for the spectrum of the result $\hat{r}$ is

$$\hat{r}[\nu] \;=\; \left(\sum_{\nu'=-\infty}^{\infty} \hat{s}[\nu']\hat{h}[\nu']\hat{\text{Ш}}[\nu-\nu'] + \hat{e}[\nu]\right)\hat{f}[\nu]\hat{d}[\nu] \tag{2}$$

where the notation $\hat{r}[\nu]$ indicates the spatial frequency ν/N, ν cycles per N spatial units, of the Fourier transform of the image r.

The Wiener filter minimizes the expected mean-square difference between the scene s and the resulting image r:

$$\begin{aligned} S^2 &= E\left\{\frac{1}{N}\int_0^N |s(x)-r(x)|^2\,dx\right\} \\ &= E\left\{\sum_{\nu=-\infty}^{\infty} |\hat{s}[\nu]-\hat{r}[\nu]|^2\right\} \end{aligned} \tag{3}$$

If the scene s and noise e are uncorrelated, stationary processes with power spectra Φ_s and Φ_e respectively, the expected mean-square restoration error can be rewritten in a form that is suitable for minimization:

$$\begin{aligned} S^2 \;=\; \sum_{\nu=-\infty}^{\infty} N \Bigg(&\sum_{\nu'=-\infty}^{\infty} \Phi_s[\nu']\hat{\text{Ш}}[\nu-\nu'] \\ &- \sum_{\nu'=-\infty}^{\infty} \Phi_s[\nu']\hat{h}^*[\nu']\hat{d}^*[\nu']\hat{\text{Ш}}[\nu-\nu']\hat{f}^*[\nu] - \sum_{\nu'=-\infty}^{\infty} \Phi_s[\nu']\hat{h}[\nu']\hat{d}[\nu']\hat{\text{Ш}}[\nu-\nu']\hat{f}[\nu] \\ &+ \left(\sum_{\nu'=-\infty}^{\infty} \Phi_s[\nu']\left|\hat{h}[\nu']\right|^2\hat{\text{Ш}}[\nu-\nu'] + \Phi_e[\nu]\right)\left(\sum_{\nu'=-\infty}^{\infty} \left|\hat{d}[\nu']\right|^2\hat{\text{Ш}}[\nu-\nu']\left|\hat{f}[\nu]\right|^2\right)\Bigg) \end{aligned} \tag{4}$$

Minimizing the mean-square error with respect to the filter transfer function values $\hat{f}[\nu]$ leads to the

definition of the optimal filter:

$$\hat{f}[\nu] = \frac{\sum_{\nu'=-\infty}^{\infty} \Phi_s[\nu']\hat{h}^*[\nu']\hat{d}^*[\nu']\hat{\text{Ш}}[\nu-\nu']}{\left(\sum_{\nu'=-\infty}^{\infty} \Phi_s[\nu']\left|\hat{h}[\nu']\right|^2 \hat{\text{Ш}}[\nu-\nu'] + \Phi_e[\nu]\right)\left(\sum_{\nu'=-\infty}^{\infty} \left|\hat{d}[\nu']\right|^2 \hat{\text{Ш}}[\nu-\nu']\right)} \tag{5}$$

This is the optimal digital filter given in Equation 26 of [1].

The mathematics of the following section is simplified by rewriting the expression for mean-square error in Equation 4 as

$$S^2 = \sum_{\nu=-\infty}^{\infty} N\left(\hat{c}[\nu] - \hat{b}[\nu]\hat{f}^*[\nu] - \hat{b}^*[\nu]\hat{f}[\nu] + \hat{a}[\nu]\left|\hat{f}[\nu]\right|^2\right) \tag{6}$$

where

$$\hat{a}[\nu] = \left(\sum_{\nu'=-\infty}^{\infty} \Phi_s[\nu']\left|\hat{h}[\nu']\right|^2 \hat{\text{Ш}}[\nu-\nu'] + \Phi_e[\nu]\right)\left(\sum_{\nu'=-\infty}^{\infty} \left|\hat{d}[\nu']\right|^2 \hat{\text{Ш}}[\nu-\nu']\right) \tag{7}$$

$$\hat{b}[\nu] = \sum_{\nu'=-\infty}^{\infty} \Phi_s[\nu']\hat{h}^*[\nu']\hat{d}^*[\nu']\hat{\text{Ш}}[\nu-\nu'] \tag{8}$$

$$\hat{c}[\nu] = \sum_{\nu'=-\infty}^{\infty} \Phi_s[\nu']\hat{\text{Ш}}[\nu-\nu'] \tag{9}$$

Then, the optimal filter transfer function $\hat{f}$ given by Equation 5 is written:

$$\hat{f}[\nu] = \frac{\hat{b}[\nu]}{\hat{a}[\nu]} \tag{10}$$

3 Imposing Spatial Constraints

In the derivation of the previous section, the Wiener filter is determined by an equation in the frequency domain:

$$\hat{a}[\nu]\hat{f}[\nu] = \hat{b}[\nu] \tag{11}$$

The spatial equivalent of this frequency domain product is the spatial convolution:

$$\frac{1}{N}\sum_{n'} a[n-n']f[n'] = b[n] \tag{12}$$

where

$$a[n] = \sum_{\nu} \hat{a}[\nu]W_N^{\nu n} \tag{13}$$

$$b[n] = \sum_{\nu} \hat{b}[\nu]W_N^{\nu n} \tag{14}$$

This convolution is equivalently expressed as a linear system of N equations in N variables (the spatial filter values). The system of equations can be expressed in matrix form:

$$\mathbf{A}\mathbf{f} = \mathbf{b} \tag{15}$$

where the $N \times N$ coefficient matrix $\mathbf{A}$ is

$$\mathbf{A}[n', n''] = \frac{1}{N} a[n' - n''] \tag{16}$$

the $N \times 1$ result matrix $\mathbf{b}$ is the array b defined in Equation 14, and $\mathbf{f}$ is the $N \times 1$ matrix of digital restoration PSF values to be determined.

In the system of equations for the Wiener filter, there are as many equations as pixels in the image. However, if the size of the spatial restoration kernel is constrained, the system of independent equations can only be as large as the number of nonzero elements in the spatial kernel. The spatially constrained kernel is designed by specifying the system of linear equations whose solution will minimize mean-square-error within the constraints.

The spatial constraint is expressed as a nonempty set of spatial locations, C, for which the restoration kernel can be nonzero. The elements that are not in the constraint set must be zero:

$$f[n] = 0 \quad \text{if} \quad n \notin C \subseteq \{0 \ldots N-1\} \tag{17}$$

If all of the points in the restoration kernel are allowed to be nonzero (i.e., $C = \{0 \ldots N-1\}$), then the optimal spatial kernel is the inverse transform of the Wiener filter (i.e., the solution of Equation 12 or 15).

The expression for MSE is defined in Equation 6 in terms of the transfer function of the optimal filter. Before this expression can be minimized with respect to the restoration kernel values, it must be expressed in terms of those elements. The filter transfer function expressed in terms of the spatially constrained kernel values is

$$\hat{f}[\nu] = \frac{1}{N} \sum_{n \in C} f[n] W_N^{-\nu n} \tag{18}$$

Substituting this expression into Equation 6, yields the MSE in terms of the constrained kernel values:

$$\begin{aligned}
S^2 &= \sum_{\nu} \left(\hat{c}[\nu] - \hat{b}[\nu] \left(\frac{1}{N} \sum_{n' \in C} f^*[n'] W_N^{\nu n'} \right) - \hat{b}^*[\nu] \left(\frac{1}{N} \sum_{n' \in C} f[n'] W_N^{-\nu n'} \right) \right. \\
&\qquad \left. + \hat{a}[\nu] \left| \frac{1}{N} \sum_{n' \in C} f[n'] W_N^{-\nu n'} \right|^2 \right) \\
&= \sum_{\nu} \hat{c}[\nu] - \frac{1}{N} \sum_{n' \in C} f^*[n'] \sum_{\nu} \hat{b}[\nu] W_N^{\nu n'} - \frac{1}{N} \sum_{n' \in C} f[n'] \sum_{\nu} \hat{b}[\nu] W_N^{-\nu n'} \\
&\quad + \frac{1}{N^2} \sum_{n' \in C} \sum_{n'' \in C} f^*[n'] f[n''] \sum_{\nu} \hat{a}[\nu] W_N^{\nu(n'-n'')} \\
&= \sum_{\nu} \hat{c}[\nu] - \frac{1}{N} \sum_{n' \in C} f^*[n'] b[n'] - \frac{1}{N} \sum_{n' \in C} f[n'] b^*[n'] \\
&\quad + \frac{1}{N^2} \sum_{n' \in C} \sum_{n'' \in C} f^*[n'] f[n''] a[n' - n'']
\end{aligned} \tag{19}$$

Minimizing with respect to the restoration kernel elements yields

$$\frac{1}{N}\sum_{n'\in C} f[n']a[n-n'] \;=\; b[n] \qquad n \in C \tag{20}$$

This is an equation with a number of unknowns equal to the number of non-zero kernel values—$|C|$. There are $|C|$ equations (differentiating with respect to each of the constrained kernel elements) in $|C|$ unknowns (the $|C|$ kernel values). This system of equations can be written as the matrix equation:

$$\mathbf{A}_C \mathbf{f}_C \;=\; \mathbf{b}_C \tag{21}$$

where $\mathbf{A}_C$ is the $|C| \times |C|$ coefficient matrix, $\mathbf{f}_C$ is the $|C| \times 1$ matrix of kernel values, and $\mathbf{b}_C$ is the $|C| \times 1$ result matrix.

The output matrix $\mathbf{b}_C$ of Equation 21 for the constrained filter is a submatrix of the corresponding matrix $\mathbf{b}$ of Equation 15 for the Wiener (unconstrained) filter. The elements of the matrix $\mathbf{b}_C$ are the elements of $\mathbf{b}$ that are in the constraint set C. Similarly, $\mathbf{A}_C$ is a principal submatrix [2] of the coefficient matrix $\mathbf{A}$ consisting only of the rows and columns of $\mathbf{A}$ named in the constraint set C.

4 Simulation Results

This section presents restoration results for artificial scenes degraded by simulated imaging devices (as described in [3]). The problem design included two variables: the width of the acquisition transfer function and the noise level. Three cases for each variable were considered, producing a total of nine experimental restoration problems. Each of the nine problems was restored with kernels constrained to a number of sizes. Then, the accuracy of the constrained restorations was compared to the accuracy of the unrestored display and Wiener restoration.

One-dimensional Fourier scenes were generated by specifying the spectral magnitude of a finite Fourier series and randomizing phase. The scene spectral magnitude $\hat{s}_\rho$ was set to

$$\hat{s}_\rho[\nu] \;=\; \begin{cases} K\exp\left(-\left(|\nu|/\alpha_s\right)^{\beta_s}\right) & \text{if } 0 < |\nu| < 2N \\ 0 & \text{otherwise} \end{cases} \tag{22}$$

with $\alpha_s = N/16$ and $\beta_s = 0.75$. Because the spectral magnitude is zero at the origin ($\hat{s}_\rho[0] = 0$), the resulting ensemble of scenes is zero-mean. The constant K was defined as 0.0704946 so that the scenes had unit root-mean-square RMS energy.

The model of the acquisition device transfer functions was suggested by Johnson[4]:

$$\hat{h}_\rho[\nu] \;=\; \exp\left(-\left(|\nu|/\alpha_h\right)^{\beta_h}\right) \tag{23}$$

All three of the transfer functions in this section are bell curves ($\beta_h = 2$).

- With $\alpha_h = 0.75$, the transfer function roll off is mostly above the Nyquist limit. This function attenuates frequency components within the Nyquist limit only slightly and will therefore cause little blurring. However, the transfer function significantly passes components above the Nyquist limit and is therefore vulnerable to aliasing.

- With $\alpha_h = 0.50$, the transfer function rolls off at a lower frequency and therefore causes somewhat more blurring, but is less vulnerable to aliasing.

- With $\alpha_{\hat{h}} = 0.25$, the transfer function is nearly zero beyond the Nyquist limit. This function virtually eliminates aliasing, but the resulting images may be blurred substantially.

Three levels of zero-mean white noise were considered. Signal-to-noise ratio (SNR) is the ratio of RMS energy of the scene to RMS energy of the noise:

$$\mathrm{SNR} \triangleq \sqrt{\frac{\sum_{\nu} |\hat{s}[\nu]|^2}{\sum_{\nu} |\hat{e}[\nu]|^2}} \tag{24}$$

For the low-noise images (high SNR), SNR=100. For the moderate-noise images, SNR=25. For the high-noise images (low SNR), SNR=5.

Real display devices are a significant component of the end-to-end imaging process but are not usually a source of much variability. Therefore, the simulated display function was not varied in these experiments—a single display model was used for all of the simulations. Schade[5] suggested a display model consisting of the sum of two Gaussian spots—the *nucleus*, a strongly-peaked central spot that contains most of the energy, and a broad *flare* spot around the nucleus. The composite display transfer function is

$$\hat{d}[\nu] = D_1 \exp\left(-\left(|\nu|/\alpha_1\right)^2\right) + D_2 \exp\left(-\left(|\nu|/\alpha_2\right)^2\right) \tag{25}$$

The parameters for the functions are taken from Schade's results: for the nucleus, $D_1 = 0.76$ and $\alpha_1 = 0.4301484$; for the flare, $D_2 = 0.24$ and $\alpha_2 = 0.0323814$. For practical reasons, the display transfer function is cut off at twice the sampling rate $\pm 2N$ (the same length as the Fourier series used to generate the artificial scenes). The effect of the truncation is insignificant.

Figure 2 illustrates the end-to-end imaging simulation for a representative scene. The top graph is the scene. Directly below it is the image created by applying the acquisition function with medium blur ($\alpha_{\hat{h}} = 0.50$) to the scene. The third graph is the sampled image. Next is the sampled scene plus moderate noise (SNR = 25). The bottom graph of Figure 2 shows the unrestored display. Acquisition blurring, aliasing due to sampling, additive sensor noise, and display degradation are all present in the output of the system. The goal of restoration is to process the noisy digital image shown in the fourth graph so that when it is displayed, the output (the bottom line) is more like the input (the top line).

The spatial kernels were constrained to have zero value at all but an odd number of locations centered at the origin—the smallest kernel, with three elements, was allowed non-zero values only where $|n| \leq 1$; the next smallest, with five elements, was allowed non-zero values only where $|n| \leq 2$; and so on. The largest constrained kernel has $(N-1)$ elements; only the element at $n = N/2$ was constrained to 0. The next-largest optimal kernel (no elements constrained to 0) is the spatial kernel of the Wiener filter.

The optimal three-point and five-point kernels for the example of Figure 2 and the corresponding transfer functions are shown in Figure 3. The Wiener filter transfer function and part of the corresponding spatial kernel are also illustrated. Only the first few elements of the Wiener kernel are shown; the magnitude of the Wiener kernel elements beyond 6 pixels from the origin is less than $0.01N$. Clearly, the optimal small kernels are quite different than the kernels produced by a truncating the Wiener PSF. As can be seen by comparing the transfer functions, the optimal three-point kernel does a fair job of approximating the Wiener filter at low frequencies but amplifies high-frequency components where SNR is lower much more than does the Wiener filter. The transfer function of the optimal five-point kernel more closely approximates the Wiener filter, but is still quite different.

Figure 4 shows the original scene, the unrestored output, the output with three-point restoration, the output with five-point restoration, and the output with Wiener restoration. Visual comparison is a subjective process, but it is clear that all of the restorations are more like the original scene than the unrestored output. It is more difficult to conclude from visual inspection which of the restorations is the best. Some of the features seem to be restored best by the three-point kernel; other features are best restored by the Wiener restoration.

Figure 5 presents numeric measures of restoration accuracy as a function of kernel size. Restoration accuracy is described by the RMS difference between the displayed image and the scene, relative to the RMS energy of the scene:

$$\text{Relative RMS Error} = \sqrt{\frac{\sum_{\nu} |\hat{s}[\nu] - \hat{r}[\nu]|^2}{\sum_{\nu} |\hat{s}[\nu]|^2}} \tag{26}$$

Each of the nine restoration problems was performed 32 times—that is, each execution used a different scene from the ensemble and different random noise. The plots show the relative RMS error averaged over all 32 executions. The standard deviations of the relative RMS error were so small that plotting them on these graphs proved impractical. The plots are shown only for kernels with 65 elements or fewer (radius 32). In all cases, only negligible improvement occurred beyond 19 elements (radius 9). (The kernel of the Wiener filter has 255 elements, a radius of 128.)

In many cases, the three-point and five-point kernels yielded results that are nearly as accurate as the Wiener filter. This is particularly true when there is little noise (e.g., SNR=100—the leftmost column). Small kernels are relatively less successful in low SNR situations (e.g., SNR=5—the rightmost column). In low SNR problems, the restoration kernel should suppress noise by local averaging, but small kernels are restricted in doing so by their size.

In the image with medium blur and medium noise ($\alpha_h = 0.50$ and SNR=25)—the middle graph of Figure 5—the average unrestored relative RMS error was 0.204613. The Wiener filter reduced this to 0.051149, a decrease of 0.1536464 or 75%. The three-point kernel resulted in an error of 0.091685, a decrease of 0.112928 or 55%. The three-point kernel (radius 1) achieved 73% of the improvement of the Wiener filter. The five-point kernel (radius 2) reduced the relative RMS error to 0.083614, a decrease of 0.120999 or 59%. This is 79% of the improvement of the Wiener filter. These small kernels achieve a large portion of the improvement of the Wiener filter with less computation.

5 Conclusion

Restoration implemented by convolution with a small kernel requires less processing than traditional Fourier-transform-based techniques such as the Wiener filter. Because convolution with a small kernel is a local operation, it is easily applied in parallel on all pixels in the focal-plane during image acquisition. The simulation results indicate that the optimal constrained restoration kernel effectively restores continuous, one-dimensional functions degraded by blurring, sampling, noise, and reconstruction—the types of degradations found in real imaging systems. Similar results were observed in simulations not presented in this paper using other one-dimensional scenes with different statistics. Two-dimensional simulations and actual restorations are in preparation.

References

[1] Carl L. Fales, Friedrich O. Huck, Judith A. McCormick, and Stephen K. Park. Wiener restoration of sampled image data: end-to-end analysis. *Journal of the Optical Society of America A*, 5(3):300–314, 1988.

[2] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, 1985.

[3] Stephen E. Reichenbach and Stephen K. Park. Computer generated scenes and simulated imaging. In *Technical Digest of the Optical Society of America Annual Meeting*, page 170, 1988.

[4] C. B. Johnson. A method for characterizing electro-optical device modulation transfer functions. *Photographic Science and Engineering*, 14(6):413–415, 1970.

[5] Otto H. Schade, Sr. Image reproduction by a line raster process. In Lucien M. Biberman, editor, *Perception of Displayed Information*, chapter 6, pages 233–278, Plenum Press, New York, NY, 1973.

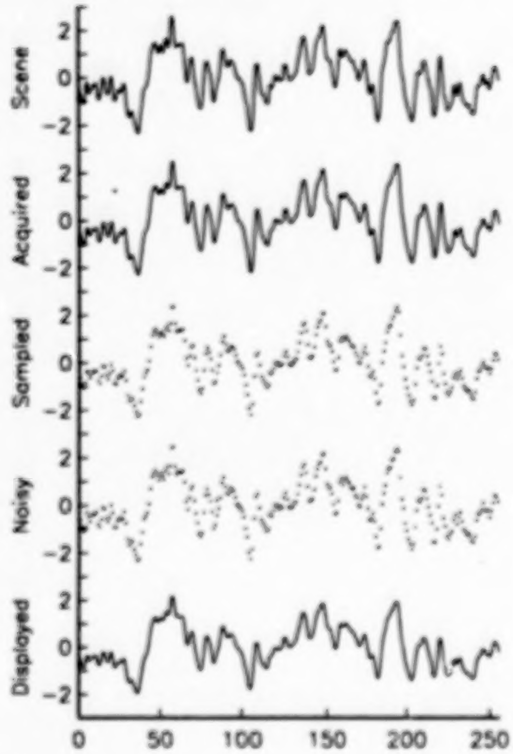


Figure 2: Simulated End-to-End Processing of a Representative Scene

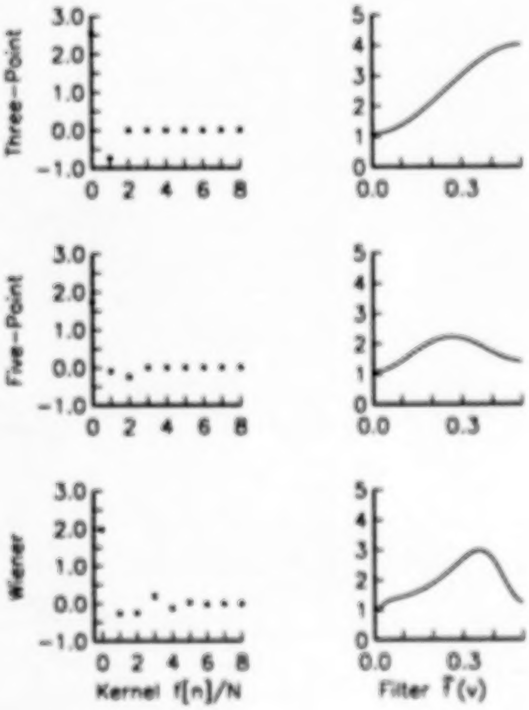


Figure 3: Restoration Functions

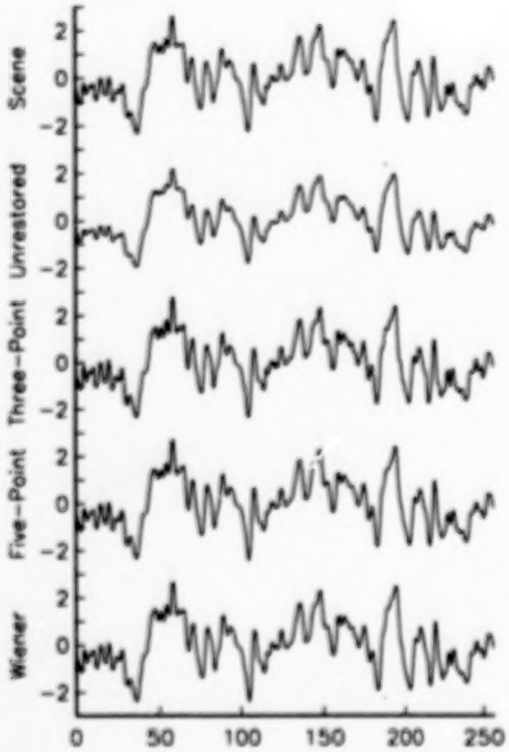


Figure 4: Representative Restoration Results

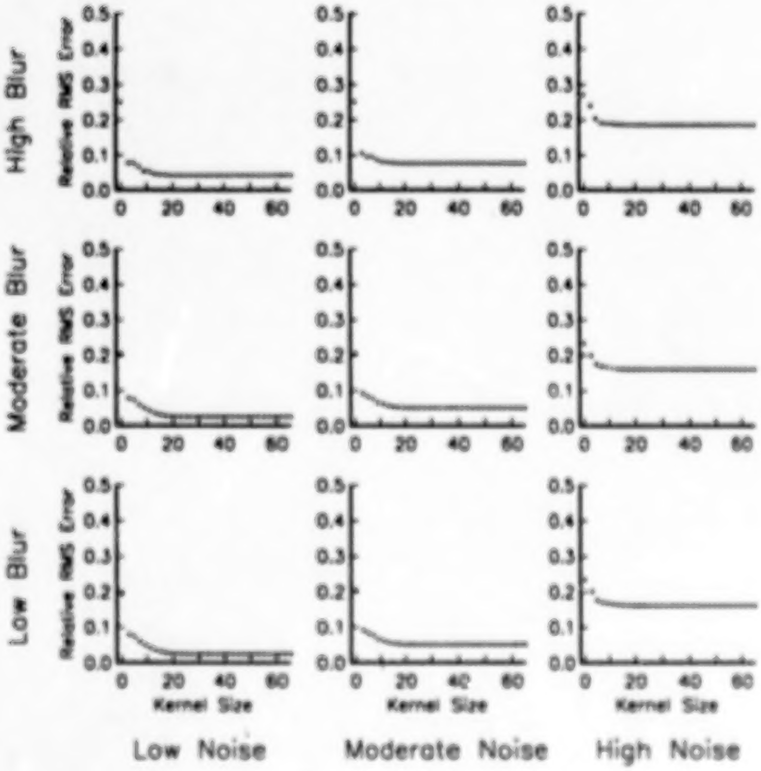


Figure 5: Relative Restoration Error

CODED-APERTURE IMAGING IN NUCLEAR MEDICINE

Warren E. Smith
The Institute of Optics
University of Rochester

Harrison H. Barrett and John N. Aarsvold
Radiology Research Laboratory and
Optical Sciences Center
University of Arizona

SUMMARY

Coded-aperture imaging is a technique for imaging sources that emit high-energy radiation. This type of imaging involves shadow casting and not reflection or refraction. High-energy sources exist in x-ray and gamma-ray astronomy, nuclear reactor fuel-rod imaging, and nuclear medicine. Of these three areas nuclear medicine is perhaps the most challenging because of the limited amount of radiation available and because a three-dimensional source distribution is to be determined. In nuclear medicine a radioactive pharmaceutical is administered to a patient. The pharmaceutical is designed to be taken up by a particular organ of interest, and its distribution provides clinical information about the function of the organ, or the presence of lesions within the organ. This distribution is determined from spatial measurements of the radiation emitted by the radiopharmaceutical.

The principles of imaging radiopharmaceutical distributions with coded apertures will be reviewed. Included will be a discussion of linear shift-variant projection operators and the associated inverse problem. A system developed at the University of Arizona in Tucson consisting of small modular gamma-ray cameras fitted with coded apertures will be described.

INTRODUCTION

In nuclear medicine a radiopharmaceutical is given to a patient. The pharmaceutical is designed to go to a particular organ of interest, such as the brain, the heart, bone, or the liver, to name a few. The three-dimensional distribution of the pharmaceutical provides clinical information about how well the organ is functioning. This is quite different than the type of information provided by x-ray imaging (electron density), magnetic resonance imaging (MRI) (proton density and magnetization relaxation rates) and ultrasound (acoustic impedance of tissue). The distribution of the pharmaceutical is determined by imaging the radiation given off by the isotope that tags it. There is always a concern to limit the total amount of radiation that the patient is exposed to, so that in nuclear medicine we have a photon-limited situation.

The isotopes used in nuclear medicine fall into two broad categories: those that emit single gamma rays directly from the nucleus, and those that emit positrons from the nucleus. Three-dimensional imaging associated with the first category is called single photon emission computed tomography (SPECT), and is the subject of this paper. In this method the photons must be blocked by attenuating apertures. Imaging associated with the second category is called positron emission tomography (PET). In PET the positron that is emitted by a source nucleus annihilates an electron within 1 to 2 mm of the source point. This annihilation results in two photons, each of approximately 511 KeV, traveling in almost opposite directions. By coincidentally detecting these two photons with spatially separate detectors, the line along their path which contains the source point can be found. This technique removes the need to physically block the photons with apertures to determine their direction of origin. With PET one can obtain extremely good resolution by nuclear-medicine standards, on the order of 5 mm. The disadvantage of PET is that an on-site cyclotron is needed to create the short-lived positron-emitting isotopes. The expense associated with this requirement has limited the number of PET facilities. SPECT imaging, on the other hand, is relatively less expensive and well established throughout the world. Thus the motivation exists to continue to improve SPECT imaging techniques to approach the quality already attainable with PET.

Two-dimensional projections of source distributions are obtained in nuclear medicine by either scanning the source in two dimensions with a single, collimated gamma-ray point detector or by forming a two-dimensional image with a camera that is capable of measuring the x and y positions of the incident gamma rays and storing them in an image histogram. Such a camera is the Anger camera, named after its inventor (ref.1). This camera can also estimate the gamma-ray energy. Energy estimation is important for rejection of Compton-scattered radiation from the nuclear-medicine image. A photon that is Compton scattered by the attenuating tissues between the source and the detector will suffer an energy shift, dependent upon the angle of scatter. Fortunately in nuclear medicine the energy spectrum of the useful isotopes is relatively narrow, so that the Compton-scattered photons can be identified and removed if their energy is outside of the peak associated with the source isotope.

Gamma rays have such a high energy that they cannot be conveniently reflected or refracted. In front of the gamma-ray camera is thus placed a shadow-casting aperture, usually made of lead or some other high atomic-number element. There are two basic types of apertures, the collimator and the pinhole. The collimator consists of a large number of usually parallel holes drilled through a thick lead plate. Each hole causes the sensitivity of a given detector element to be confined to a narrow pencil that intersects the source distribution. This narrow pencil is an approximation to a line integral through the source. All of the holes together form a parallel-line 2-D projection of the source onto the 2-D detector. The pinhole is a single hole punched in a relatively thin lead plate. This aperture produces a pinhole image of the source distribution on the 2-D detector. The pinhole image represents a series of line integrals through the object that converge on the pinhole. Conventional systems in nuclear medicine often employ parallel-hole collimators. The coded-aperture systems to be discussed employ arrays of pinholes.

Tomography in nuclear medicine is achieved by taking multiple views of the source distribution, and reconstructing a 3-D estimate of the source from these views. Conventionally these views are obtained by rotating a large gamma-ray camera fitted with a parallel-hole collimator around the patient. The camera stops every few degrees and takes a two-dimensional snapshot of the patient lasting about a minute. Each snapshot of the patient approximates a set of parallel line integrals, defined by the collimator, through the source volume at the particular angle. Neglecting attenuation of the source by the body, the set of all of these snapshots over 180 or 360 degrees constitutes an approximation to the Radon transform of the source distribution. The inverse Radon transform (ref.2) is then applied to these projections to form an estimate of the three-dimensional source distribution. This inverse involves filtering and then back-projecting each projection into the reconstruction space, and can be done rapidly with modern equipment. Without modification of the inverse Radon transform to include attenuation of the photons by the body, reconstructions appear darker for pixels deeper within the tomographic slice. This attenuation problem can be corrected analytically by the attenuated Radon transform (refs. 3,4), assuming constant attenuation and a known convex attenuation boundary. Typical scan times for the rotating-camera approach are 30 to 45 minutes. Dynamic studies of pharmaceutical uptake are ruled out because of the required motion of the camera about the patient.

In this paper we discuss tomography in nuclear medicine with non-moving coded apertures. The reconstruction of both 2-D and 3-D source distributions will be addressed. A coded-aperture system for nuclear medicine being developed at the University of Arizona will be described.

CODED-APERTURE TOMOGRAPHY IN NUCLEAR MEDICINE

In nuclear medicine we are able to observe only a small number of photons because the radiation dose to the patient is kept as low as possible and because the fractional solid angles of the collimator or pinhole openings are small, on the order of 10^{-5}. These openings must be small because in shadow-casting the ability of the aperture to resolve two closely spaced points in the source is directly proportional to the size of the openings. Thus we have a fundamental trade-off between the signal-to-noise ratio (SNR) in the nuclear-medicine image, which goes as the square root of the number of detected photons, and the resolution of the system. This trade-off is quite different in focusing systems, such as lenses focusing visible light, where the diffraction-limited spot size decreases (thus improving resolution) as the aperture is opened up, allowing more photons into the system.

There is thus strong motivation for increasing the number of photons in a nuclear-medicine image without degrading the resolution. To this end coded apertures have been developed. Figure 1 shows a single-view coded-aperture system. Here a planar source distribution is projected through an aperture consisting of several pinholes to form a coded image. The position of the pinholes represent the code. We have thus increased the number of photons detected by the system, at the price of overlap in the pinhole views of the object. This overlap is referred to as spatial "multiplexing", and is more serious for larger objects and denser spacing of the pinholes. Thus we suspect immediately that the code should be optimized with respect to the type of object that we wish to view.

In this planar case, neglecting radiometry and obliquity factors, we can write the coded image as a convolution of the source with the aperture:

$$g(x'',y'') = f(x''/m,y''/m) ** h(x''/M,y''/M) \qquad (1)$$

where the double prime indicates detector coordinates. The quantity $g(x'',y'')$ is the coded image, $h(x''/M,y''/M)$ is the scaled aperture function, and $f(x''/m,y''/m)$ is the scaled source distribution. The source scaling $m = (z-d)/z$ and the aperture scaling $M = d/z$, where z is the source-aperture distance, and d is the source-detector distance. The two-dimensional convolution operator is represented by $**$. As we see, both the source and the aperture functions are scaled differently in forming the coded image.

To form a reconstruction of the original source distribution, we use the concept of matched filtering. A matched filter is a version of the actual signal that we are looking for. It can be shown that a matched filter is the optimum filter to be used to detect a signal in the presence of noise (ref. 5). In the coded-imaging case, the matched filter is a properly scaled, inverted, complex-conjugated version of the original code, so that the reconstruction $\hat{f}(x'',y'')$ can be written as

$$\hat{f}(x''/m,y''/m) = g(x'',y'') ** h^*(-x''/M,-y''/M). \qquad (2)$$

Equation (2) can be written, using Eq. (1), as:

$$\hat{f}(x''/m,y''/m) = f(x''/m,y''/m) ** [h(x''/M,y''/M) ** h^*(-x''/M,-y''/M)], \qquad (3)$$

where the bracketed term represents the overall point-spread function (PSF) of the data-taking and reconstruction process, and is called the autocorrelation of the code. We must design the code to make its autocorrelation function as close to a delta function as possible, simultaneously allowing as many openings as possible. Unfortunately, these two requirements work against each other. Generally the autocorrelation has a large central peak surrounded by a background with structure that depends upon the number of openings. This background tends to both smear out the reconstruction as well as increase the noise in the reconstruction.

Note that we are not restricted to real and positive functions in our search for optimum codes. Any physically realizable code will be real and positive because of the shadow-casting nature of the image formation from the incoherent source. Bipolar complex codes can be simulated, however, by creating 4 separate codes and forming 4 separate coded images and suitably adding them in the computer with proper positive, negative, and imaginary weights. We pay the price for this flexibility by increasing the amount of time needed to form an image, however.

There has been considerable research into defining codes to optimize the SNR of the final reconstruction. Some of the more well-known codes are random pinhole arrays (ref. 6), the Fresnel zone plate (ref. 7), the annulus (ref. 8), and time-modulated apertures (ref. 9). Much of this code optimization has been in the context of single-view imaging of a planar object, however, as in Fig. 1. If we were to image a three-dimensional volume object with this approach, our reconstruction of Eq. (3) would be for a particular plane of the source, depending upon the scale factor used for the matched filter. The other planes of the source would present a strong background superimposed on this reconstruction, degrading both resolution and SNR of the plane of interest. Thus there is a fundamental limitation of the planar correlation decoding method described above because our basic data set consisting of a single view is not complete enough. We must in fact take multiple views of a volume object so that we are sampling its three-dimensional Fourier components sufficiently. Combining multiple views of the object to form a single volume reconstruction is not obvious with the planar decorrelation method described. In fact, we must generalize our entire approach to the problem and move away from the shift-invariant formulations of Eqs. (1-3).

With a multiple-view system, shown schematically in Fig. 2, we must give up the convenience of shift invariance. Thus the convolution operation can no longer be used to connect the object to the data. Instead the mapping from object to data takes on the more general form:

$$g(x'',y'',z'') = \int_{\text{source}} f(x,y,z)\ h(x'',y'',z'';x,y,z)\ d^3V, \qquad (4)$$

where $g(x'',y'',z'')$ represents all of the coded images (spread out in three-dimensions), $f(x,y,z)$ is the three-dimensional source distribution, $h(x'',y'',z'';x,y,z)$ is the shift-variant mapping from the source to the coded images, and d^3V is a volume element of the source. All of the radiometry and aperture geometry is contained within $h(x'',y'',z'';x,y,z)$. The distribution $g(x'',y'',z'')$ forms the data set from which to reconstruct the estimate of the object $\hat{f}(x,y,z)$. Numerically, it is necessary to map the continuous problem into a discrete formulation by choosing a suitable basis set. We can see how this is done by a demonstration with a one-dimensional analog of Eq. (4):

$$g(x'') = \int_{\text{source}} f(x)\ h(x'';x)\ dx. \tag{5}$$

We can approximate f(x) and g(x") each in the following way:

$$f(x) \approx \sum_{n=1}^{N} f_n\ \psi_n(x) \tag{6}$$

$$g(x'') \approx \sum_{m=1}^{M} g_m\ \phi_m(x'') \tag{7}$$

where

$$f_n \equiv \int_{-\infty}^{+\infty} f(x)\ \psi_n^*(x)\ dx \tag{8}$$

and

$$g_m \equiv \int_{-\infty}^{+\infty} g(x'')\ \phi_m^*(x'')\ dx''. \tag{9}$$

The basis sets $\psi_n(x)$ and $\phi_m(x'')$ span their respective spaces and are assumed orthonormal in this development. Thus we have approximated the source and the data with N and M discrete coefficients, respectively. An example of a particular source basis set is the "pixel" basis set, where the $\psi_n(x)$ are N non-overlapping shifted and scaled rectangle functions. Another example is the Fourier basis set, where the $\psi_n(x)$ represent complex exponentials, the eigenfunctions of shift-invariant operators. By the appropriate substitutions, we can now approximate Eq. (5) as:

$$g_m \approx \sum_{n=1}^{N} h_{mn}\ f_n \tag{10}$$

where

$$h_{mn} \equiv \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} h(x'';x)\ \psi_n(x)\ \phi_m^*(x'')\ dx\ dx''\ . \tag{11}$$

Thus the shift-variant problem of Eq. (5) is represented by a matrix multiplication, where the n^{th} column of the matrix **H**, with elements

h_{mn}, represents the discrete, m-element shift-variant PSF due to the n^{th} expansion term of the source. In fact, if we choose the $\psi_n(x)$ and $\phi_m(x")$ correctly, dependent upon h(x";x), we can have $h_{mn} = h_m$ for m = n, and $h_{mn} = 0$ otherwise. In other words, **H** is diagonal or pseudo-diagonal if M is not equal to N. This choice of basis leads to what is called the singular value decomposition (SVD) of **H**.

Generalizing to three dimensions, utilizing basis functions such as $\psi_n(x,y,z)$ and $\phi_m(x",y",z")$, we can write Eq. (4) for all coded images in a way identical to Eq. (10). This can be done by simply ordering the N expansion coefficients of f(x,y,z) into a one-dimensional N x 1 vector **f** and the M expansion coefficients of g(x",y",z") into a one-dimensional M x 1 vector **g**:

$$\mathbf{g} = \mathbf{H}\,\mathbf{f} + \mathbf{n} \;, \tag{12}$$

where we have introduced the M x 1 zero-mean noise vector **n** to allow for image degradation from effects outside of the direct mapping due to **H**. Equation (12) is the general form of a shift-variant imaging system that we will use in the subsequent discussion of finding the source estimate $\hat{\mathbf{f}}$.

In general, Eq. (12) represents an ill-posed problem, in that one or more of the following conditions occur: no $\hat{\mathbf{f}}$ exists that satisfies **g** exactly; $\hat{\mathbf{f}}$ is not unique; the solution $\hat{\mathbf{f}}$ is sensitive to small changes in **g** or **H**. We must usually content ourselves with a solution $\hat{\mathbf{f}}$ that agrees with **g** to within some limits, and if these approximate solutions are not unique, choose one that satisfies some independent prior knowledge about **f**. There are several techniques for finding $\hat{\mathbf{f}}$, such as singular value decomposition (SVD) alluded to briefly above (ref. 10), Monte Carlo methods (ref. 11), linear estimation theory (refs. 12, 13), and iterative methods (ref. 14). We will focus here on the Monte Carlo method, which we have found to be a practical technique for handling the large-scale pseudoinversion of Eq. (12) in the coded-aperture context. We have successfully simulated the reconstruction of volume objects **f** of up to 32000 source elements from data sets **g** consisting of nearly the same number of detector elements using less than 10 Mbytes of computer memory, in CPU times under 30 minutes on a VAX 8600. The reason for this space and time economy is that the **H** matrix is sparse in coded-aperture imaging. Of course this sparseness is reduced as the number of pinhole openings increases, or as the size of the pinholes increases, since more detectors are being illuminated by each source element.

In the Monte Carlo reconstruction process we define an energy function E that is minimized when the reconstruction $\hat{\mathbf{f}}$ achieves a desired level of agreement with the data **g** and simultaneously is consistent with any prior knowledge about the types of sources present. Such prior knowledge in the nuclear-medicine context consists of source positivity, source boundary, and perhaps correlation statistics between nearby source pixels. One of the cost functions that we have used is:

$$E = (1-\alpha)\,||\ \mathbf{g} - \mathbf{H}\,\hat{\mathbf{f}}\ ||^2 + (\alpha)\,||\ \hat{\mathbf{f}} - \langle\hat{\mathbf{f}}\rangle\ ||^2 \tag{13}$$

where the double bar indicates magnitude of the vector and the brackets indicate an averaging process over nearest-neighbor pixels in the given estimate $\hat{\mathbf{f}}$. The first term of this expression measures agreement with the data, and the second term imposes a smoothing constraint on $\hat{\mathbf{f}}$, relating each pixel of the reconstruction to its nearest neighbors. The adjustable scalar α weights the agreement-with-data term against the smoothing term. We begin the reconstruction process with an initial guess at $\hat{\mathbf{f}}$ (a zero object or a uniform grey-level object). We then perturb each pixel of $\hat{\mathbf{f}}$ and calculate ΔE, the perturbation to E. This calculation is relatively rapid, because only a few detectors out of the hundreds or thousands of detectors actually see the perturbation to $\hat{\mathbf{f}}$. It should be mentioned that only non-zero elements of $\mathbf{H}$ are required, so that even a 32000 by 32000 matrix can be stored in a small fraction of the space otherwise needed.

The perturbation is always accepted if $\Delta E <= 0$, and if $\Delta E > 0$, it is accepted according to the Boltzmann probability of statistical mechanics:

$$P(\Delta E) = \exp\,(-\Delta E/kT) \tag{14}$$

where k is Boltzmann's constant (usually set to 1 in this context) and T is an effective "temperature" of the estimate at any given time. If T is large, we frequently allow large positive ΔEs into the reconstruction. If T is small, the probability of accepting large positive ΔEs is much reduced. The concept of starting the reconstruction at a large T and slowly reducing its value as E is decreased is known as "simulated annealing" (ref. 15). Such annealing is necessary if the energy surface E exhibits local minima: the occasional uphill energy swings of the reconstruction reduce the probability of being trapped in a local energy minimum. For quadratic energy functions as shown in Eq. (13), annealing is not required. However, if E is not quadratic, perhaps due to the imposition of strongly non-linear prior knowledge, annealing may become significant in improving the reconstruction. We have observed the importance of annealing for cases of very powerful prior knowledge, such as binary-object reconstruction, when a pixel is constrained to be on or off and the rules weighting its agreement with neighboring pixels are very nonlinear.

In our experience with the Monte Carlo algorithm, we find that we can typically obtain estimates $\hat{\mathbf{f}}$ that agree very well with the data, within a fraction of a percent. The smoothing constraint is a very important one; without it we get good data agreement, but there are large local fluctuations in the reconstruction that reduce its visual quality. The smoothing operation is imposed continuously as the reconstruction evolves permitting an ongoing compromise between the smoothing constraint and the data constraint.

An important aspect of coded-aperture imaging is the determination of the system operator **H**. This matrix contains all of the geometry and radiometry (including attenuation, assuming known source-volume attenuation parameters) mapping the discrete object space to the discrete detector space. **H** can be modeled theoretically as in Eq. (11), but for an actual system it should be found experimentally by a calibration procedure. Such a procedure consists of placing a point-source gamma-ray emitter in a volume attenuator that approximates the expected attenuating properties of the source, and stepping the point source through this volume one pixel location at a time. For each pixel location, the data set corresponds to a column of the **H** matrix, including the effects of attenuation, radiometry, aperture vignetting, and detector efficiencies. It is important to have a bright enough source so that the SNR of the **H**-matrix elements is high enough not to degrade the SNR of the reconstruction. Reconstructing the object using this **H** matrix automatically includes the effects of attenuation and detector characteristics.

There are several advantages to pinhole coded-aperture imaging as compared to the conventional rotating collimated gamma-ray camera. The ability of a collimator to resolve two source points degrades faster with source depth than with a pinhole aperture. Thus the coded-images may contain higher spatial-frequency information than the collimator images. Also, the number of photons detected by a coded-aperture with many openings is greater than that of a collimator because the fractional solid angle of the coded aperture is greater. Thus we expect the SNR of a coded image to be superior to that of a collimator image. Finally, in a coded-aperture system consisting of multiple views, no detector motion is required so that dynamic studies are possible.

There are also disadvantages to the coded-aperture approach. Even though we detect more photons, this advantage is offset by the fact that we suffer from the multiplexing problem in the data sets. These two effects are coupled and both together determine the final SNR of the reconstruction. An additional complication is the need to carefully characterize the **H** matrix through the calibration procedure described above. For a fixed system of modules and attenuation boundaries, however, this need be done only periodically. The attenuation boundaries can be fixed by placing the patient within a water sleeve, whose outer dimensions remain fixed. The reconstruction of the object from a coded-image data set is also more difficult in general than applying the inverse Radon transform in conventional tomography, but special-purpose hardware is being developed to optimize this procedure.

A set of small, independent gamma-ray cameras are being developed at the University of Arizona for applications in coded-aperture imaging (ref. 16). These cameras use a 10 cm by 10 cm NaI crystal coupled optically to 4 photomultiplier tubes (PMTs). The outputs of the 4 PMTs form a 20-bit address that extracts from a previously defined lookup table the statistically most likely x and y location of the gamma-ray impact point on the crystal face. Each camera, or a bank of cameras, has its own coded aperture, thus forming a camera module. These modules can then be positioned about the patient in a configuration that will optimally utilize the detector area. Figure 3 is an example of an 8-view system for planar tomography that is currently being constructed in Arizona to be used for heart and brain imaging.

Preliminary simulations with systems similar to that of Fig. 3 demonstrate that state-of-the-art reconstructions are obtainable with data-acquisition times of the order of a third or less than that of the conventional rotating gamma-ray camera, which are typically 30 to 40 minutes. This potential data-acquisition time reduction, as well the static nature of the system allowing dynamic studies, may contribute to improving the state-of-the-art in nuclear-medicine imaging.

CONCLUSION

We have briefly described the principles of imaging in nuclear medicine, and have focused on a particular approach using coded apertures. The formulation of this shift-variant problem was developed, and a particular reconstruction algorithm was presented. A coded-aperture system being developed at the University of Arizona for tomographic imaging in nuclear medicine was briefly described.

ACKNOWLEDGMENTS

This work was supported by the National Cancer Institute through grant no. 2 P01 CA 23417. We thank Bruce Moore for his technical assistance.

REFERENCES

1) Anger, H.O., "Scintillation Camera," Rev. Sci. Instrum., *29*, 27 (1958).

2) Barrett, H.H., and W. Swindell, *Radiological Imaging: The Theory of Image Formation, Detection, and Processing*, Vols. I and II (Academic, New York, 1981).

3) Tretiak, O., and C. Metz, "The Exponential Radon Transform," SIAM. J. Appl. Math. *39*, 341 (1980).

4) Clough, A.V., and H.H. Barrett, "Attenuated Radon and Abel Transforms," J. Opt. Soc. Am. A, *73*, 1590-1595 (1985).

5) Gaskill, J.D., *Linear Systems, Fourier Transforms, & Optics* (John Wiley, New York, 1978).

6) Dicke, R.H., "Scatter-hole Cameras for X-rays and Gamma Rays," Astrophys. J. *153*, L101 (1968).

7) Barrett, H.H., "Fresnel Zone Plate Imaging in Nuclear Medicine," J. Nucl. Med. *13*, 382-385 (1972).

8) Simpson, R.G., "Annular Coded-Aperture System for Nuclear Medicine," doctoral dissertation (University of Arizona, Tucson, Ariz., 1978).

9) Koral, K.F.,W.L. Rogers, and F.G. Knoll, "Digital Tomographic Imaging with a Time-Modulated Pseudorandom Coded Aperture and an Anger Camera," J. Nucl. Med. *16*, 402 (1975).

10) Strang, G. *Linear Algebra and Its Applications*, (Academic, New York, 1976).

11) Smith, W.E., R.G. Paxman, and H.H. Barrett, "Image Reconstruction from Coded Data: I. Reconstruction Algorithms and Experimental Results," J. Opt. Soc. Am. A, *2*, 491-500 (1985).

12) Melsa, J.L., and D.L. Cohn, *Decision and Estimation Theory*, (McGraw-Hill, New York, 1978).

13) Smith, W.E., and H.H. Barrett, "Linear Estimation Theory Applied to the Evaluation of *A Priori* Information and System Optimization in Coded-Aperture Imaging," J. Opt. Soc. Am. A, *5*, 315-330 (1988).

14) Frieden, B.R., "Image Enhancement and Restoration," in T.-S. Huang, ed., *Picture Processing and Digital Filtering*, Vol. 6 of Topics in Applied Physics, (Springer-Verlag, New York, 1975).

15) Kirkpatrick, S., C.D. Gelatt, Jr., and M.P. Vecchi, "Optimization by Simulated Annealing," Science, *220*, 671-680 (1983).

16) Aarsvold, J.N., H.H. Barrett, J. Chen, A.L. Landesman, T.D. Milster, D.D. Patton, T.J. Roney, R.K. Rowe, R.H. Seacat, III, and L.M. Strimbu, "Modular Scintillation Cameras: A Progress Report,"Medical Imaging II: Image Formation, Detection, Processing, and Interpretation, SPIE *914*, 319-325 (1988).

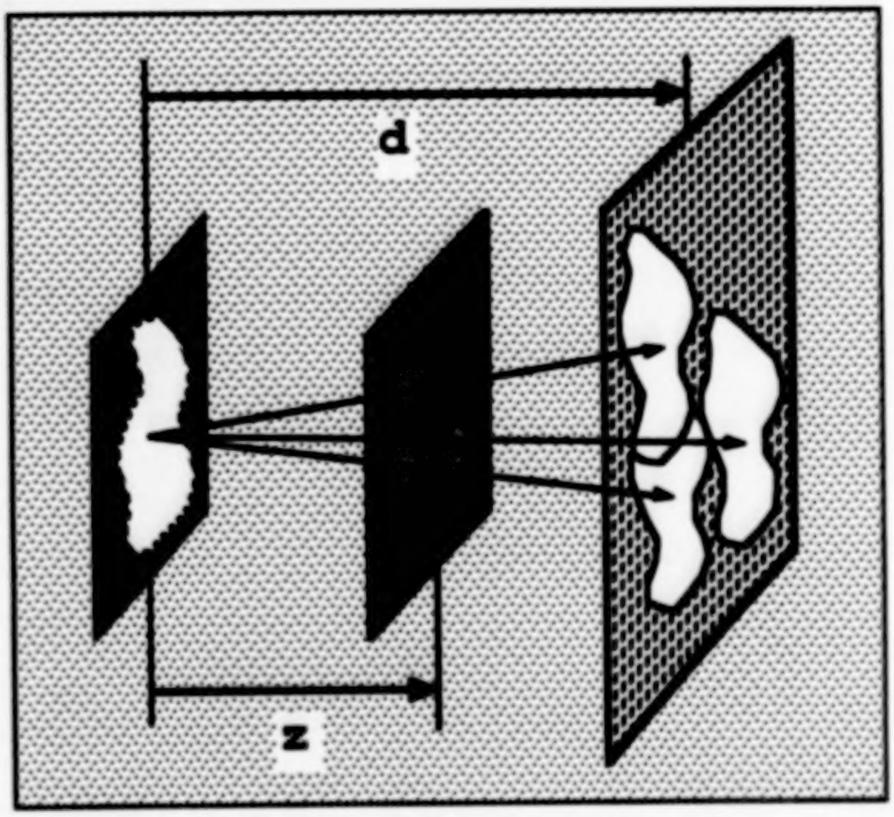


Figure 1) A single-view coded-aperture system, imaging a planar source.

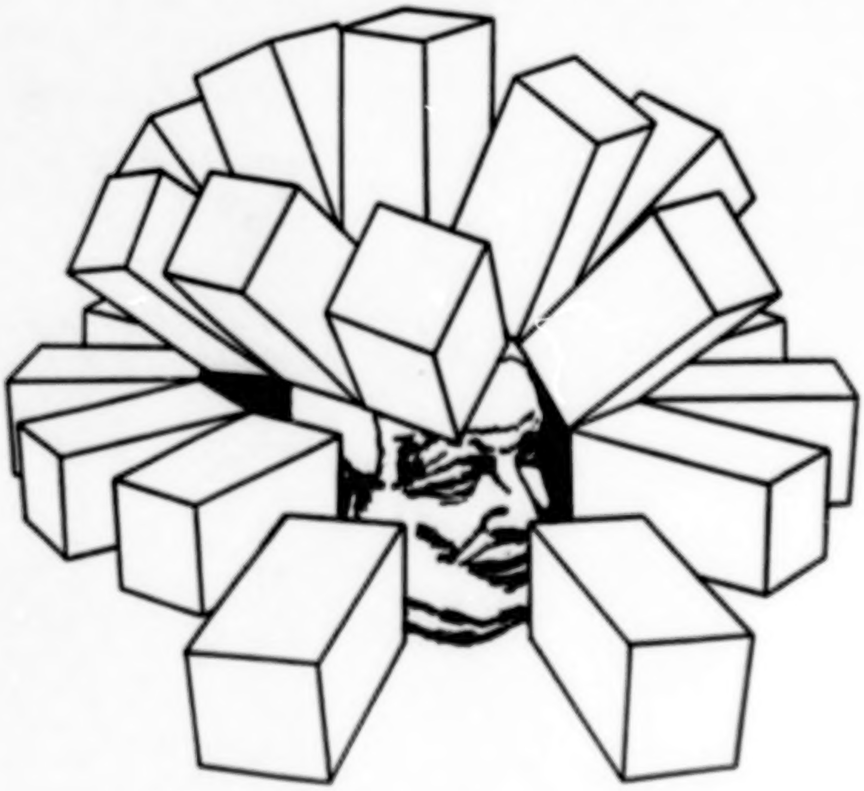


Figure 2) A multiple-view coded-aperture system, imaging a volume source.

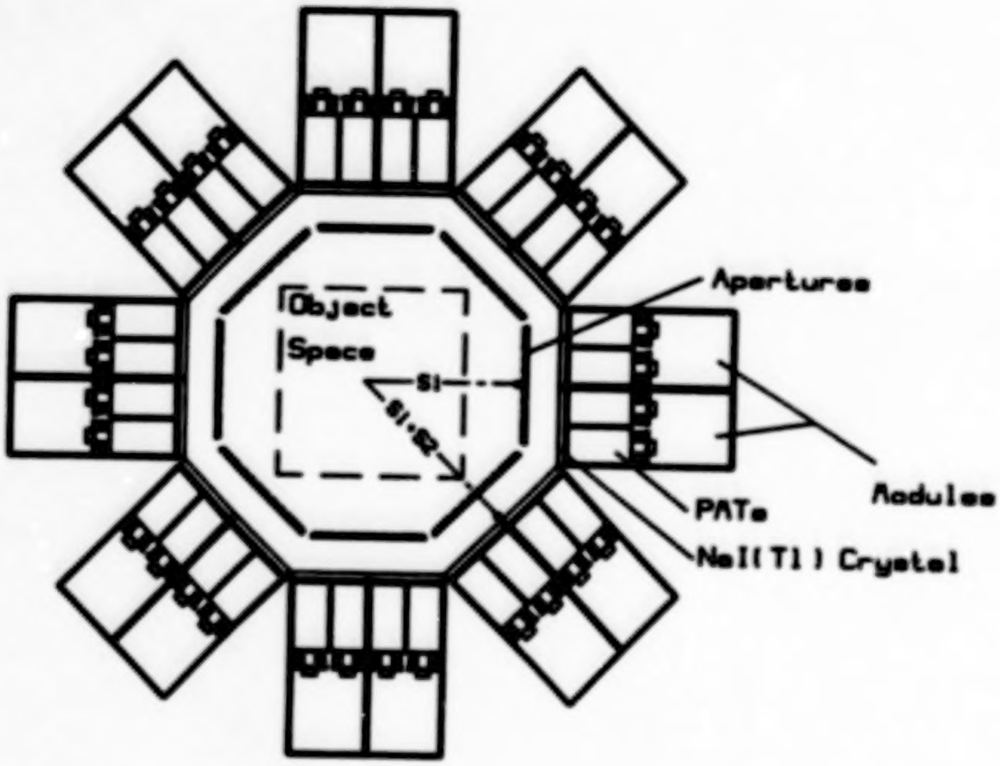

Object
Space
S1
S1+S2
Apertures
Modules
PMTs
NaI(Tl) Crystal

Figure 3) An octagonal coded-aperture system being used at the University of Arizona for planar tomography.

CONTEXT DEPENDENT ANTI-ALIASING IMAGE RECONSTRUCTION

P. R. Beaudet

Westinghouse Electric Corporation
Advanced Technology Division
P.O. Box 1521, MS 3D12
Baltimore, MD

A. Hunt and N. Arlia
Westinghouse Research and Development Center 601-1B44C
1310 Beulah Road
Pittsburgh, PA

ABSTRACT

Image Reconstruction has been mostly confined to context free linear processes; the traditional continuum interpretation of digital array data uses a linear interpolator with or without an enhancement filter. In this paper, anti-aliasing context dependent interpretation techniques are investigated for image reconstruction. Pattern classification is applied to each neighborhood to assign it a context class; a different interpolation/filter is applied to neighborhoods of differing context.

It is shown how the context dependent interpolation is computed through ensemble average statistics using high resolution training imagery from which the lower resolution image array data is obtained (simulation). A quadratic least squares (LS) context-free image quality model is described from which the context dependent interpolation coefficients are derived.

It is shown how ensembles of high resolution images can be used to capture the a priori spacial character of different context classes. As a consequence, a priori information such as the translational invariance of edges along the edge direction, edge discontinuity, and the character of corners is captured and can be used to interpret image array data with greater spatial resolution than would be expected by the Nyquist limit. A Gibb-like artifact associated with this super-resolution is discussed. More realistic context dependent image quality models are needed and a suggestion is made for using a quality model which now is finding application in data compression.

I. INTRODUCTION

The work presented in this paper builds upon theory[1] that was developed further at the Westinghouse Advanced Technology Laboratory and more recent work at the Westinghouse Research and Development Center. The goal of this work is to develop optimal adaptive methods of interpreting image data. By matching the interpretation function to the local characteristics of the scene, a context dependent interpreter is designed which offers superior performance over context independent interpolation functions such as bilinear and cubic convolution.

When applied to sampled image data, this context dependent interpolation function yields an image which is free of the aliasing artifacts caused by image frequency content too high for the sampling frequency. Because of its ability to recognize familiar patterns in the sampled data before its interpretation, anti-aliasing interpolation selects the interpretation which is most probable given the a priori knowledge of context class patterns. This interpretation process is sometimes referred to as super-resolution.[2] The benefits of super-resolution are

- o Provides a method of contextually and artificially increasing the sampling frequency from which the known system modulation transfer function can be better compensated.
- o Gives a better procedure for image zoom.
- o May lead to adaptive methods of image gathering such as is provided in nature through eye movement and neural pre-processing

In Section II, a distinction is made between data interpretation vs data interpolation. Here, rationale is given for pursuing this work and the basic theoretical approach is given. In Section III, an experiment is described designed to show what benefits might be expected from super-resolution. In Section IV, results are presented of context dependent interpretation and some of the resulting artifacts are discussed. The discussion continues in Section V where a basis for future work is provided and a discussion of a more realistic quality model is presented.

II. INTERPRETATION VS INTERPOLATION

Images are often defined by their fourier content.[3]

$$I(\mathbf{x}) = \iint_{-\infty}^{\infty} \frac{d^2\mathbf{k}}{(2\pi)^2} I_{\mathbf{k}} \, e^{i\mathbf{k}\cdot\mathbf{x}}$$

where $\mathbf{x}$ and $\mathbf{k}$ are 2-vectors, $I(\mathbf{x})$ is the image and $I_{\mathbf{k}}$ is its fourier transform. A sampled image data set can be written as the sampling of an image at integer (or periodic) valued of $\mathbf{x} = \mathbf{i}$. (We select $\Delta x = \Delta y = 1$ throughout this paper.)

$$I(\mathbf{i}) = \iint_{-\infty}^{\infty} \frac{d^2\mathbf{k}}{(2\pi)^2} I_{\mathbf{k}} \, e^{i(\mathbf{k}\cdot\mathbf{i})} = \sum_{\mathbf{m}} \iint_{-\pi}^{\pi} \frac{d^2\mathbf{k}}{(2\pi)^2} I_{(\mathbf{k}+2\pi\mathbf{m})} \, e^{i(\mathbf{k}\cdot\mathbf{i})}$$

The sampled "image" generated by frequency $\mathbf{k}$ of unit amplitude and that generated by $\mathbf{k} + 2\pi\mathbf{m}$ for any integer 2-vector $\mathbf{m}$ are the same; A fundamental frequency $\mathbf{k}$ is indistinquishable from any of its aliasing frequencies $\mathbf{k} + 2\pi\mathbf{m}$. As a consequence, it is really impossible to determine the true frequency content of an image without some a priori knowledge. The typical engineering assumption that is made is that the true image has no frequencies larger than the Nyquist frequency $|k_x| < \pi$ and $|k_y| < \pi$.

This assumption is most often wrong and forcing it to be correct by placing a smoothing filter in the image gathering process may result in loss of information (interpretable spacial resolution). It does simplify the display process, however, which is then unambiguous.

To get a better appreciation of the information loss that is possible, consider the images of figures 1a and 1b. Here, a road, pipe, cable or other narrow-object illuminates a diagonal set of pixels which if interpreted according to the Nyquist assumption would lead to a display (interpretation) consisting of a string of blobs one for each diagonal pixel. It is a goal of this work to interpret this data more as a human might do as illustrated in Figure 1b. What other assumption than Nyquist could be used to better accomplish this human-like interpretation of the data? Surely it is that almost everywhere in a scene there is some direction of minimal spacial frequency, while its orthogonal direction may have a very high frequency content; frequencies even higher than the Nyquist frequency.

By data interpretation,[4] it is meant the generation of a continuous image function $I(\mathbf{x})$ from a sampled subset $I(\mathbf{i})$ taking into consideration the directions in the image data over which there are minimal/maximal changes. Such a process is context dependent because the interpolation process depends upon the scene patterns. In contrast, by data interpolation it is meant a context free interpolation of the data such as is provided by the sinc function interpolation based upon the Nyquist assumption.

$$I(x,y) = \sum_{i,j.} I(i,j)\,\mathrm{sinc}\,(\pi|x-i|)\,\mathrm{sinc}(\pi|y-j|).$$

To implement a context dependent interpretation process, the neighborhood of each data sample must be classified into one of many context classes, K. This can be done by computing the local gradient and classifying based upon gradient magnitude and direction. More complex classes are also possible for images containing lines, line ends, corners, etc. For each context class, K, the interpolation formula

$$I(x,y) = \sum_{\mathbf{i}} I(i,j)\, g_K(x-i,y-j)$$ is used where g_K is the interpolation

coefficients (function) matched to the context class. It is anticipated that such a process would be capable of distinguishing the line-like objects of figure 1 and provide for a more human-like interpretation. The extent to which this is possible is the subject matter of this paper.

III. THE Experiment

An experiment was designed to determine the extent to which a priori knowledge of scene content could be used to improve sampled data interpretability. A real high resolution television picture was taken with a CCD camera, digitized to obtain a 512x384 digital image, and averaged over sixteen frames to reduce noise. It was an image of a white piece of rectangular cardboard tilted by about 30° relative to the camera axias (see figure 2). 15x12 blocks of pixels were averaged to obtain 180 coarse images of the scene, each 32x32 pixels. These 180 coarse images varied in the manner (phase) by which the data were averaged from the finer resolution data. The scenes were contextually classified as illustrated in Figure 3, and one of the coarse images was contextually interpreted to achieve the high resolution image shown in Figure 4. The philosophy for deriving the super-resolution interpretation functions is subsequently presented. This philosophy uses a least squares image quality model which is believed to be at the heart of the Gibbs-like[5] artifacts seen in Figure 4.

The interpreted function for each context class, K, is expressed as

$$I(x,y)=\sum_{i,j,\in N} I(i,j)\, g_K(x-i,y-i) \quad .$$

where N was selected as a 5x5 neighborhood centered at the pixel nearest the point x,y. The fine grid (512x384) was used for discrete points within each pixel.

The interpolated image was compared to the original (ground truth) data in a least squares manner yielding a cost function:

$$C = \langle (I(x,y) - \tilde{I}(x,y))^2 \rangle_K$$

Here $\langle\ \rangle_K$ is an ensemble average over all 180 images at all defined context class K centers. To obtain the "optimal" interpolation function, we simply take a functional variation of C with respect to g_K (x-i, y-j) giving a system of normal equations which decouple for each subpixel location x,y. The system of normal equations is

$$\sum_{i,j\in N} \langle I(i,j)I(i',j')\rangle_K\, g_K(x-i',y-j') = \langle I(x,y)I(i',j')\rangle_K$$

For each x,y and K, this is a system of twenty five equations for the contextual interpolation coefficients $g_K(x-i,y-j)$ to be applied at the 5x5 array in the neighborhood of each pixel classified as K to achieve the interpolation value I(x,y). The Matrix $\langle I(i,j)I(i',j')\rangle$ and vector $\langle I(x,y)I(i',j')\rangle_K$ are ensemble averages over the 180 processed coarse images and the original fine resolution image.

IV. Results

The results of this first experiment is shown in Figure 4. A context free bilinear interpolated image is shown in Figure 6. Clearly, the context dependent process retains the translational invariance along the edge and is much "sharper" than the context free bilinear interpolator which also shows the staircase aliasing artifact. But Figure 4 has a Gibb-like artifact which in itself is a distraction. Surely as humans, we wouldn't interpret the coarse data shown in Figure 5 with these Gibb-like oscillations! Where do they come from?

To understand the results of this experiment, all we really need to recognize is that the data are noisy.

A sharp edge discontinuity with a white noise background should have to be filtered a-la Weiner[6] if based upon a least squares fidelity criteria. The Weiner filter is a very sharp filter and will essentially truncate all spatial frequencies whose amplitude is below the noise level while preserving all those above the noise level. Figure 7 illustrates the response of an edge function to such a process. The high frequency truncation does a best-least squares job but results in the Gibb oscillations. It is believed that this Gibb's phenomena is the process at work here and results from the implicit assumption that errors near edges are just as important as those away from the edge; (least squares criteria).

V. Discussion and Future Work

The Gibbs artifact can be subdued if a different model to an image quality measure is taken. The sharp discontinuity of an edge, together with the least squares criteria, places a severe restriction upon the approximating function and admits short but large excursions from the edge function. Another image quality measure may prove more effective in yielding an edge approximation which is more in keeping with a human interpretive approach. It is a quality model which is finding application in DPCM data compression. The model considers human toleration to a slight "jitter" of the pixel (sometimes called rate distortion). Any error in the approximating function is compared not just to the expected noise variance, σ^2, but to the noise variance plus rate distortion. If $\Phi(x)$ is the approximating function to $I(x)$, then $(I(x)-\Phi(x))^2$ must be compared to

$\sigma^2 + h^2 \left| \frac{d\Phi}{dx} \right|$ where h is the variance equivalent subpixel jitter.

The modified cost function is then $\left\langle \dfrac{(I(x) - \Phi(x))^2}{\sigma^2 + h^2 \left(\frac{d\Phi}{dx} \right)^2} \right\rangle$.

This is a nonlinear functional which in the limit of very low contrast edges (noisy edges) leads to the previous least squares measure. But for large contrasts, the model compares the error to the slope of the approximating function. In this high contrast case, an edge gets approximated by the exponential function $\Phi(x) = 1-e^{-\alpha x}$ which distributes the representation error uniformly and provides the type of solution one might expect a human to choose. This edge function is shown in Figure 8. It trades off a more uniform transition in exchange for a sharper drop near the edge. The solution is forced to be a smooth transition because if

$\frac{d\Phi}{dx} = 0$ anywhere, then any error there is given a very large weight.

References

1. Beaudet, P. R., "Context Dependent Interpolation" Image Science Mathematics Symposium. November, (1976), Monterey, California. Editors Carrol O. Wilde, Eamon Barrett.

2. Marple, S. L., "Digital Spectral Analysis with Applications" Chapter 16, Prentice Hall, 1987.

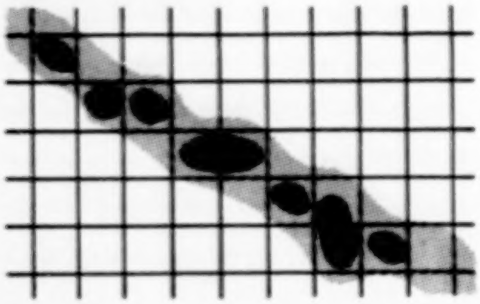
3. Rosenfeld, A., Kak, A. C. "Digital Picture Processing" Academic Press, 1982.

4. Beaudet, P. R., "Context Sensitive Modeling", Proceedings of the ERIM International Symposium on the Remote Sensing of the Environment, Anne Arbor, Michigan, May 1981.

5. Wylie, C. Ray "Advanced Engineering Mathematics" McGraw Hill, 1975

6. Wiener, Norbert "Time Series" The M.I.T. Press, 1964

- **Context-Free (Aliasing)**



- **Context-Dependent (Anti-Aliased)**

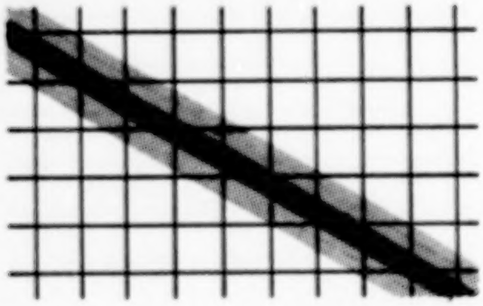


Figure 1a. Aliasing Artifacts (What is Aliasing)

Example:

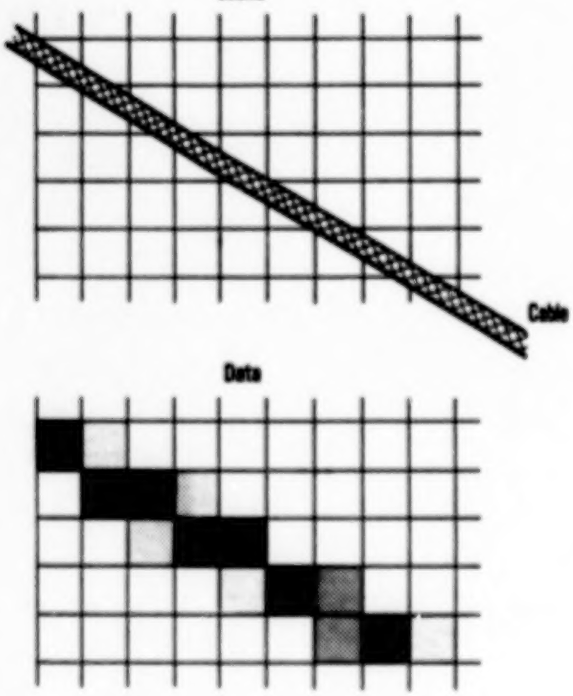


Figure 1b. Data Interpretation (Display)

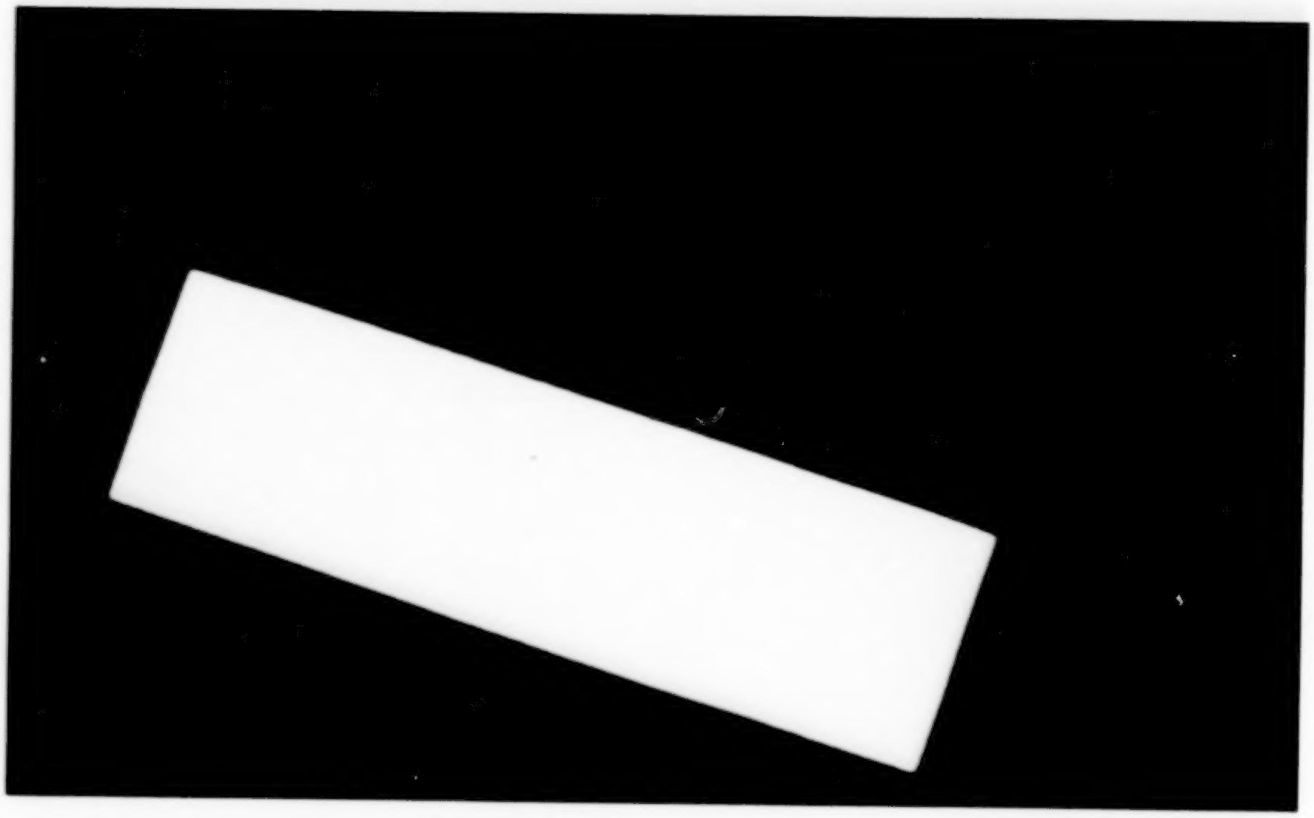


Figure 2. Original TV Image

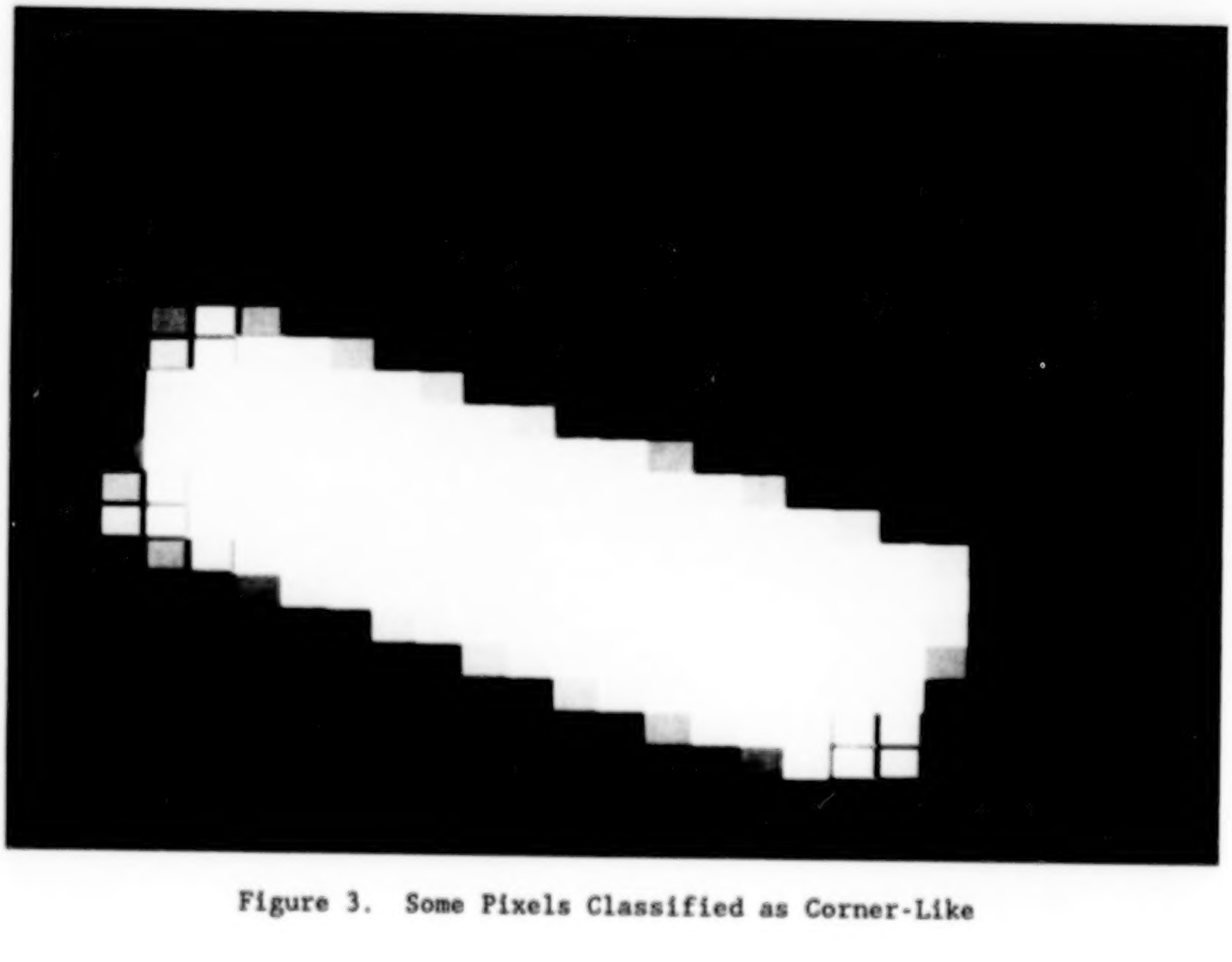


Figure 3. Some Pixels Classified as Corner-Like

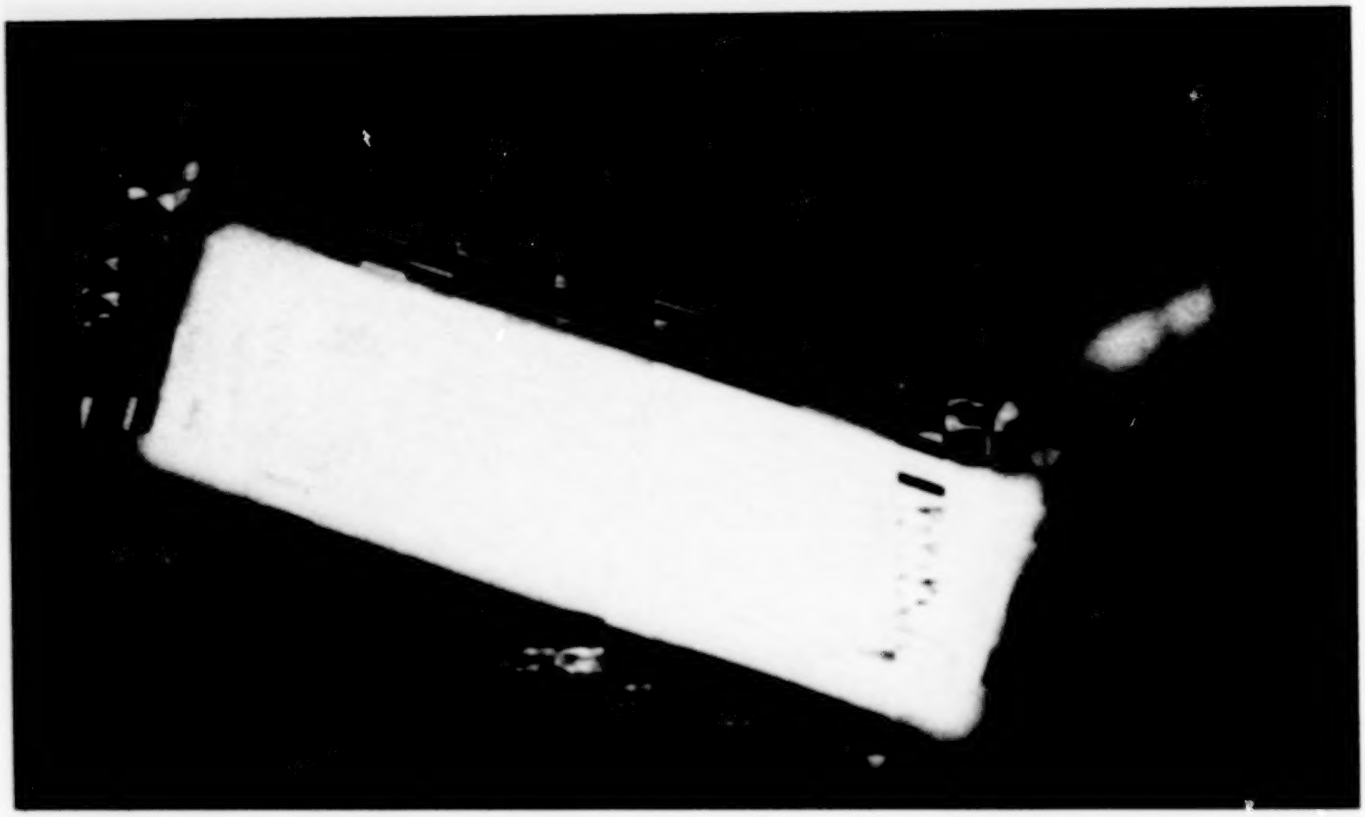


Figure 4. Contexturally Interpreted Coarse Data

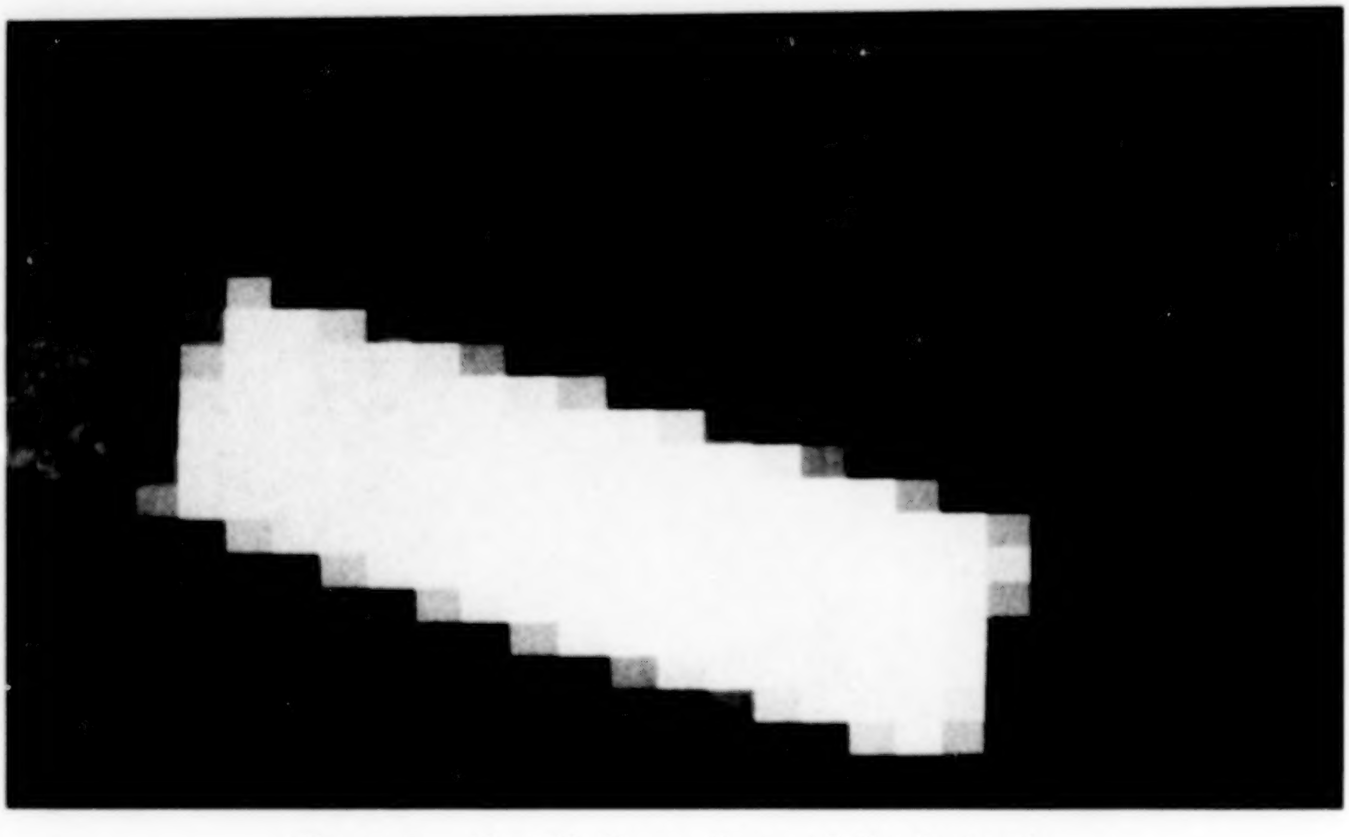


Figure 5. 32 × 32 Coarse Scene of the Rectangle

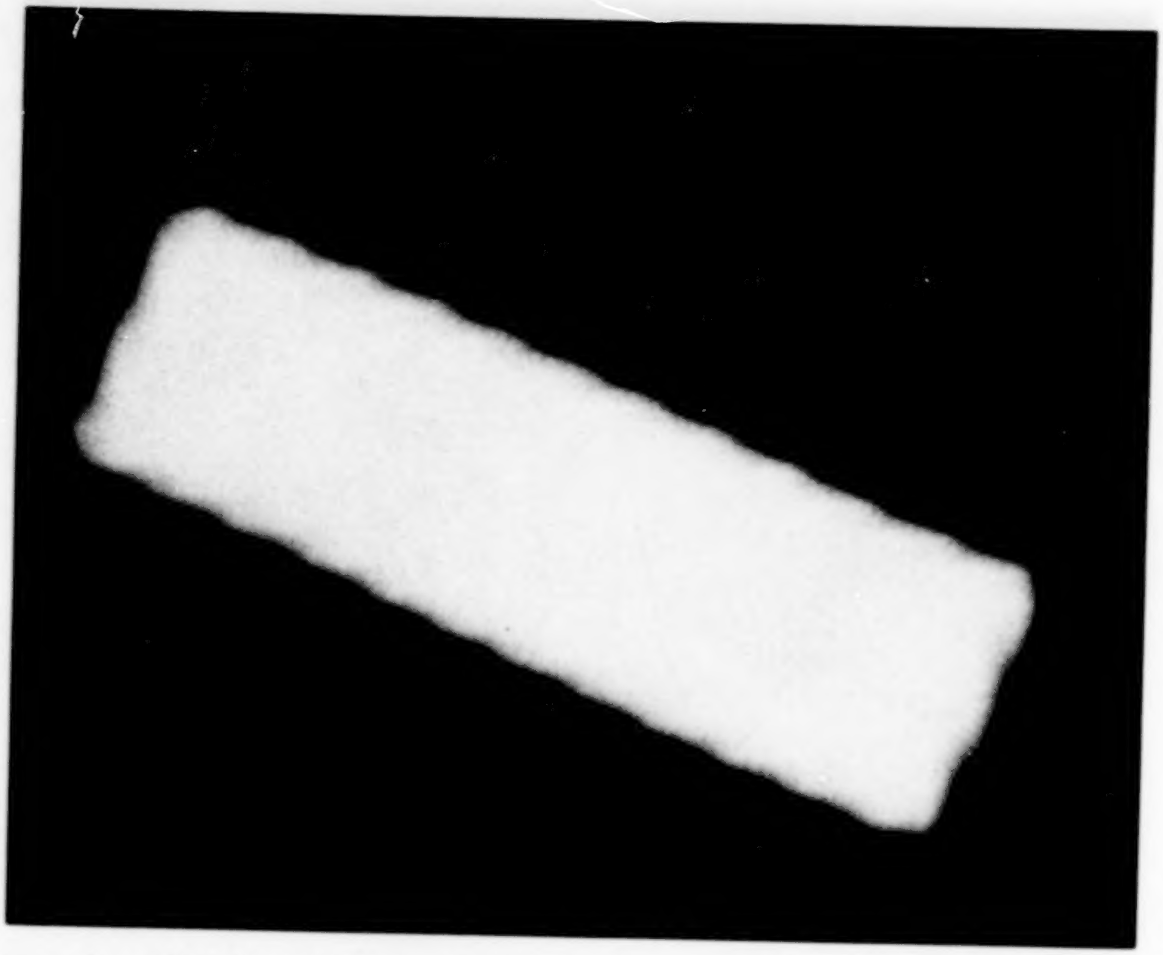


Figure 6. Aliasing Artifacts Caused By Context-Free Bilinear Interpolation

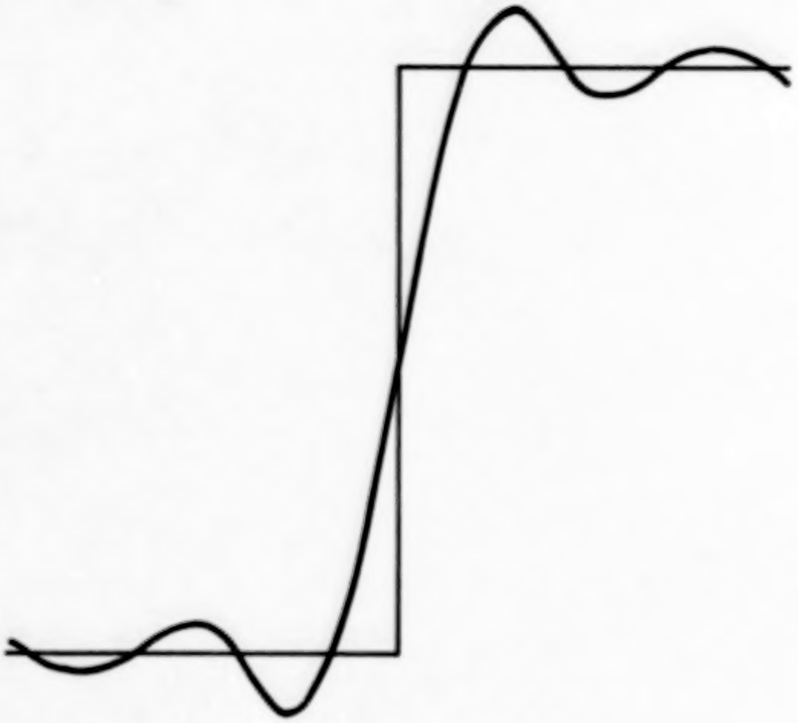


Figure 7. Gibb-Like Oscillations Caused By Least Square Error Criteria and Sharp Truncation in the Fourier Plane

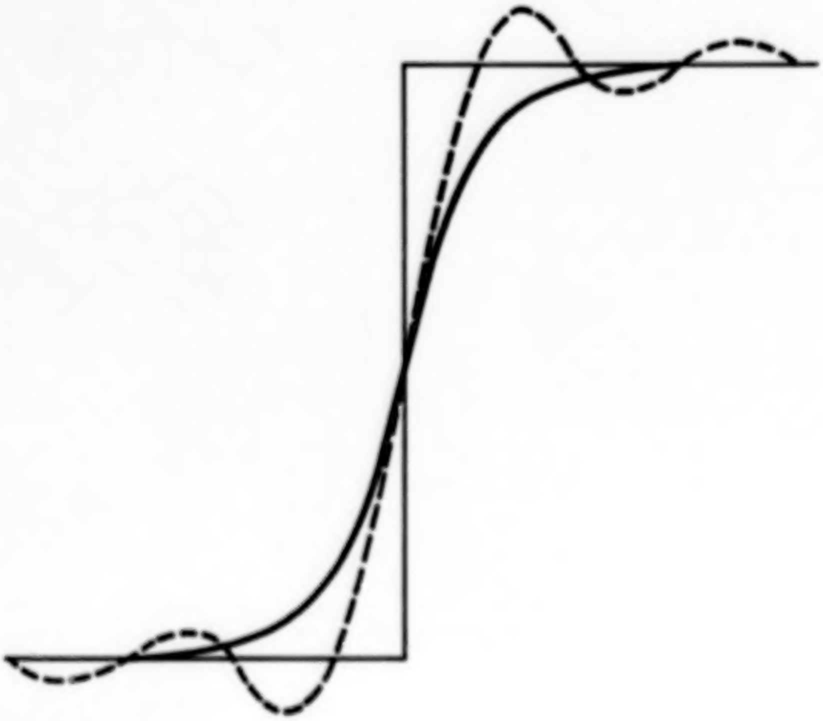


Figure 8. Exponential Edge Interpretation Based Upon a Modified Image Quality Model

Context Dependent Prediction

and

Category Encoding for DPCM Image Compression

Paul R. Beaudet
Westinghouse Electric Corporation
Advanced Technology Division
Baltimore, MD

ABSTRACT

Efficient compression of image data requires the understanding of the noise characteristics of sensors as well as the redundancy expected in imagery. Herein, the techniques of Differential Pulse Code Modulation[1] (DPCM) are reviewed and modified for information-preserving data compression.

The modifications include:
- o Mapping from intensity to an equal variance space
- o Context dependent one and two dimensional predictors
- o Rationale for nonlinear DPCM encoding based upon an image quality model
- o Context dependent variable length encoding of 2x2 data blocks
- o Feedback control for constant output rate systems

Examples are presented at compression rates between 1.3 and 2.8 bits per pixel. The need for larger block sizes, 2D context dependent predictors, and the hope for sub-bits-per-pixel compression which maintains spacial resolution (information preserving) are discussed also.

Introduction

This paper discusses an image data compression technique which has shown great promise in studies performed at Westinghouse. The technique incorporates "context dependent" and "category encoding" methods. The major steps of the process comprise:

1. equal variance mapping of the input data
2. context dependent prediction of the current pixel value
3. nonlinear differential pulse code modulation (DPCM) of the pixel differences
4. variable length encoding of blocks of the DPCM deltas
5. error correction encoding to protect against transmission and storage bit errors.

Ref. 1. Azriel Rosenfeld and Avinash C. Kak, Digital Picture Processing, Chapter 5, Academic Press, Inc., 1982

A block diagram of the basic concept is given in Figure 1. The input data is passed through a nonlinear function selected to make the noise variance of the signal independent of the absolute signal level. The exact shape of this function is tailored to compensate for signal dependent noises; at low signal levels the predominant noise is system noise, while at higher levels shot noise or scene noise predominate. The equal variance mapping technique renders the DPCM process equally effective at all signal levels.

The context dependent predictor (CDP) applies a set of coefficients to neighboring pixels (casual process) to predict the next pixel intensity. The choice of coefficients depends upon the pattern classification assigned to the neighborhood. The CDP can be either one or two dimensional. The predicted pixel is subtracted from the actual pixel data to obtain a "delta".

DPCM is the third step in the data compression process. A graded, symmetrical table of delta is used. The DPCM code keeps the noise of the delta code selection an approximately constant percentage of the actual signal gradient, and produces an output data set whose probability density function is sharply peaked. This latter property makes the output deltas particularly suited to variable-length encoding techniques such as Huffman encoding.

The next step is the variable-length encoding of 2x2 data blocks. The data is encoded in blocks of two by two pixels for encoding efficiency. Variable-length encoding assigns the output Huffman code to the 2x2 blocks of deltas in such fashion that the shortest groups represent the most frequently used delta blocks, and the longest groups represent the most infrequently used delta blocks. This minimizes the number of bits in the output data stream.

Because the data compression process removes almost all redundancy from the signal, an error in signal transmission can cause a large loss of data. This makes it necessary to follow the data compression process with an error-correcting encoding process. This process reintroduces a small amount of signal redundancy with high efficiency, so that an encoding overhead of only one or two percent suffices to reduce bit loss to an acceptably low level.

Equal Variance Mapping

The equal variance mapping function is derived from a noise model.

The noise model used assumed three independent noise sources:

1. electronic noise
2. shot noise
3. scene noise

Scene noise can be viewed as a fluctuation in scene reflectivity which is not considered as containing any significant information. The purpose of the equal-variance mapping function is to map the input intensity I into a new variable X such that the noise, σ_x, in this new variable is independent of the absolute level X. Figure 2 illustrates this property of the function.

The noise model expresses the input intensity variance,

$$\sigma_I^2 = \underbrace{\sigma_o^2}_{\text{electronic noise}} + \underbrace{\Gamma I}_{\text{shot noise}} + \underbrace{(\beta I)^2}_{\text{scene noise}}$$

In this model, $N = I/\Gamma$, measures the input signal in electrons. In terms of N,

$$\sigma_N^2 = \left(\frac{\sigma_o}{\Gamma}\right)^2 + N + (\beta N)^2$$

σ_o/Γ is the number of electrons of electronic noise. Some sensor electrometers have reduced this noise component down to seven electrons. More realistically, electronic noise of hundreds of electrons might be expected. $\sqrt{N}$ is the shot noise due to the random nature of photon emission. A fraction, β, of the scene signal is considered texture or scene noise. Typically, $\beta \simeq .01$ is used. The equal variance mapping function satisfies the differential equation

$$\boxed{\sigma_x = \sigma_I \frac{dx}{dI}}$$

which has solution

$$X(I) = \Phi \frac{\sigma_o}{\beta} \log \left[\frac{\left(\sqrt{\sigma_o^2 + \Gamma I + \beta^2 I^2} + \beta I + \frac{\Gamma}{2\beta}\right)}{\left(\sigma_o + \frac{\Gamma}{2\beta}\right)} \right]$$

Φ is selected so that $X(I_{MAX}) = X_{MAX}$. We often select $X_{MAX} = 4095$ so as to optimize the dynamic range of a 12 bit processor.

Context Dependent Predictor

One dimensional predictors are used mostly because two dimensional ones require sensor equalization which makes all of the detectors respond similarly to input intensity; many systems cannot afford the luxury of data equalization. A context-free linear predictor is of the form

$$X_{i+1} = X_i + \epsilon(X_i - X_{i-1})$$

where ϵ is a constant ($\epsilon = .75$ is sometimes used). A context-dependent predictor has the same "form" of this equation, but $\epsilon(x_i - X_{i-1})$ is interpreted as "a function of" $(X_i - X_{i-1})$. This function is selected to maximize the prediction accuracy and can be determined using scene statistics. The results suggest that an equivalent multiplier, $\in$, should be negative for very small differences, $\in=1$ for intermediate but significant gradients and $\in=.7$ for very large gradients. Intuitive rationale supports these findings; in uniform regions, small differences should be smoothed (negative $\in$), and $\in=1$ near large edge discontinuities would overshoot the edge.

Two dimensional predictors are currently being inserted into the context dependent DPCM image compression software. Also, the encoding block is being increased from 2x2 to 4x4 pixel blocks. The two dimensional predictor uses data from a causal neighborhood as shown in Figure 3. This data set is used to characterize the neighborhood in terms of a context class index, k. The set of prediction coefficients used is dependent upon k but is otherwise linear:

$$\tilde{X} = \sum_{i,j \in C_k} \alpha^{(k)}{}_{ij} \; X_{ij}$$

The DPCM Processor

The block diagram of the DPCM processor is given in Figure 4. The input to the process is the output of the equal-variance mapper. The predictor functions in this mapped space, attempting to predict the next pixel value as accurately as possible. The better the predictor, the narrower will be the density function of the resulting deltas. The processor includes gain in the forward and reverse paths so that a dynamic range adjustment (DRA) can be made by the feedback parameter Q. This controls the output bit rate and ensures optimum delta encoding of the data in the transmission mode. The delta values are selected from a table of possible deltas which has been derived by consideration of data noise and rate distortion. This consideration leads to a hyperbolic sine relationship between the actual delta difference and the delta code. A typical delta table is given in Table 1.

Variable-Length Encoding with Blocking

The distribution of δ-code values is peaked near zero. If P_i is the probability of the $i^{th}\delta$-code (relative frequency), then a variable length code (Huffman code) having about $b_i \approx -\log_2 P_i$ bits for the i^{th} δ-code state minimizes the number of bits required to encode the image values. Without encoding, the bits per pixel might be 5 since δ-codes range over $-15 \leq +15$; (5 bits per pixel (bpp)). Because P_0 is typically .7 or so, only 0.5 bits are allocated to that state. As a consequence, an encoding inefficiency is incurred whenever a single state occurs with such high probability. To avoid this encoding inefficiency, 2X2 blocks are Huffman encoded with 2 bits $\cong -\log_2(.7)^4$ assigned to blocks consisting of all zero δ-code states.

To provide a structure which can be continuously processed, the 2x2 blocks are arranged as shown in Figure 5. To further improve the encoding, contextual information from neighboring blocks is used to create an adaptive Huffman encoding scheme. The sign bits from the delta codes of the pixel on either side of the block, taken together with two bits from the adjacent top pixels are used to define 16 context states. Statistics have been accumulated for all of these states and a different Huffman encoding table is used for each context.

Category Encoding Method

It is impractical to assign variable length codes to each of the possible 2x2 block states. Since 5 bit DPCM codes are used, this would require storage for $2^{4x5} = 2^{20}$ code words for each of 16 contextual situations. To avoid such a massive table, category (cluster) encoding is used. Many of the 2^{20} states which are clustered into separate categories include only one of the block states while other categories include many of these states. For those categories containing many states, an additional resolution code must be supplied in order to identify the specific state of the cluster.

In the schema shown in the Figure 6, a 3-bit pixel category (sub-category) is assigned. The eight states correspond to δ-code values +0, ±1, ±2, ±(3, 4, 5, 6 or 7) and $|\delta\text{-code}| \geq 8$. The 3-bit pixel categories are used to define a 5-bit α, β pair category (1x2 category) according to the strategic table shown. Note that if both α and β pixels have δ-code values between ±2, the 1x2 category code completely determines the state of both pixels; no resolution code need be appended to the category. When the two upper and lower α, β pair 1x2 category codes are brought together, each pair contributes five bits to a variable length 2x2 category code table. These 10 bits plus the four context bits constitute an address for a single 16K ROM containing the complete adaptive Huffman encoding tables. This is a thousand times less storage than the direct approach!

Error-Correction Overhead

Error correction encoding processes operate by the insertion of error correction bits into blocks of data before subjecting the data to a noisy process such as storage or transmission. After the data is transmitted (or read from storage) a mathematical operation on the combined block of data plus error correction bits reveals the presence and location of any bits in error, up to a limit determined by the number of correction bits and the size of the data block. The quality of the code is measured by the maximum number of bits in error, L, that can without fail be corrected by the code within a fixed block. The overhead ratio M/K of this process is the number of error correction bits M divided by the number of bits K in the original data block. This ratio is a function of the ratio L/K. For a fixed value of this latter ratio, the bit error rate improves with the block size K in a complex manner. An analysis of this problem was made assuming an output bit error rate of 10^{-15} with an input bit error probability of 10^{-5} and an overhead ratio of just over 1 percent (1.14%) was required. Under these assumptions, calculations showed that, with a block size of 5000, any 5-bit error could be corrected. This is far better performance than needed for any perceived application.

Compression Performance

The performance of this data compression technique is dependent on the statistics of the input images. Our experience to date has been with scenes taken in the visible spectrum. With this type of input, compression from 2 to 2.5 bits per pixel can be achieved with a noise increase of no more than 3 dB. Images compressed to 1.3 bpp show no loss of information content. Figure 7(b-d) is the compressed data of a scene at (1.4, 1.7, and 2.1 bpp). There is little perceptual difference between this compressed image and the original, even at 1.3 bpp. Figure 8 (b-d) shows another image compressed at 1.5, 2.0, and 2.4 bpp. The two images were courtesy of the National Institute of Health, Bethesda, Maryland.

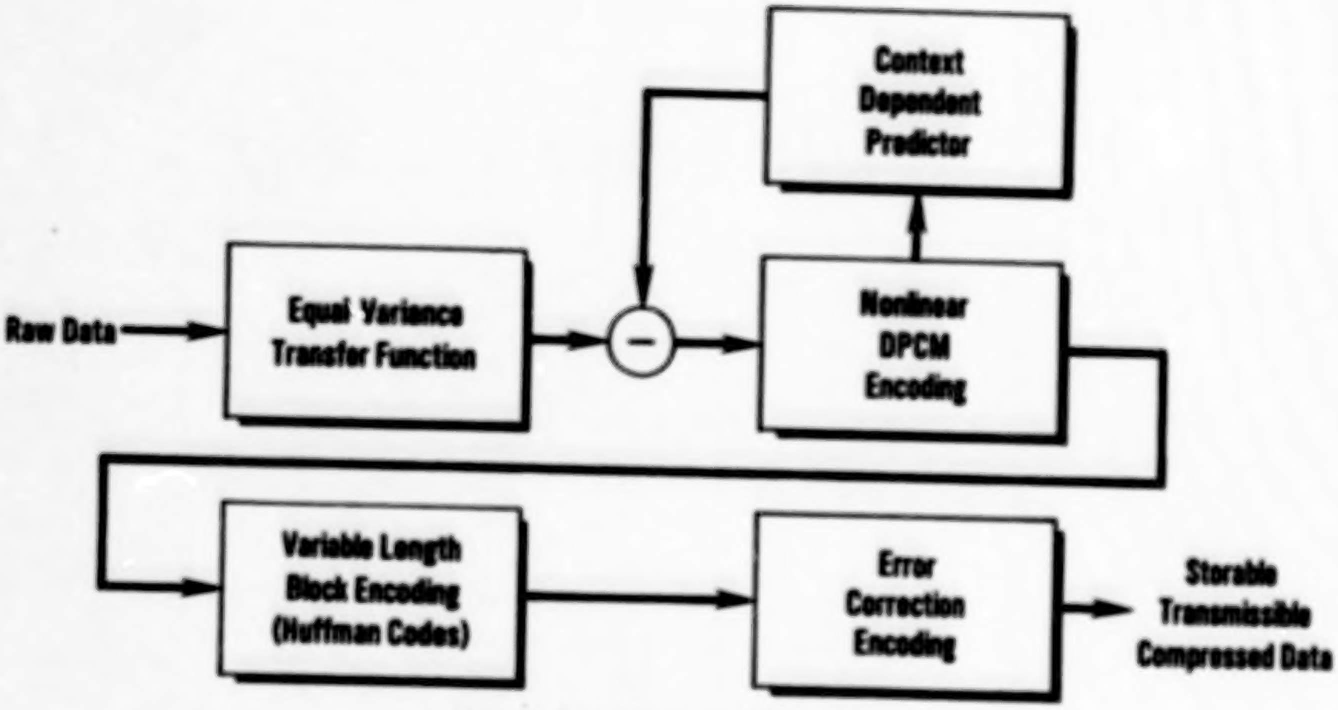
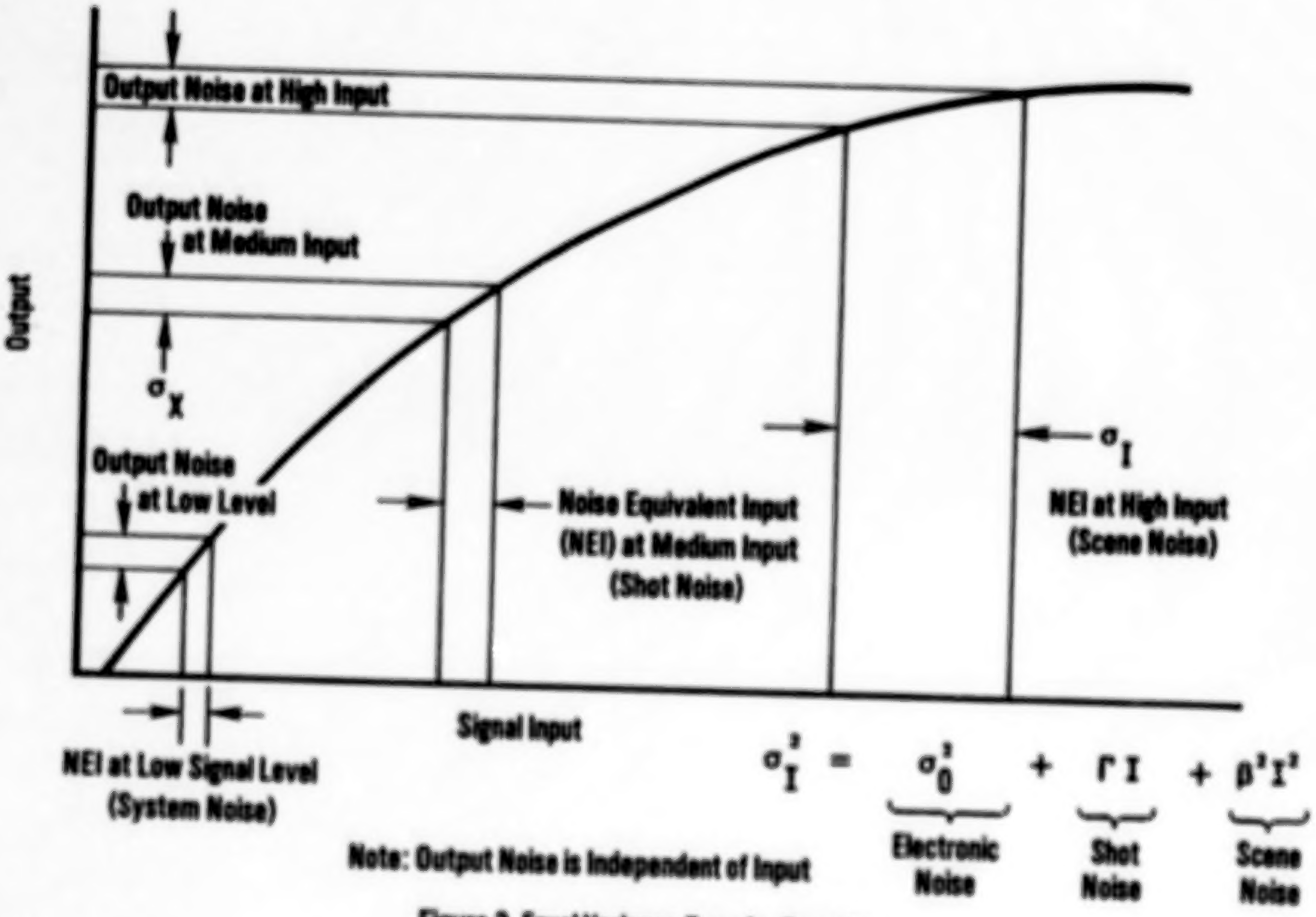

Raw Data
Equal Variance Transfer Function
−
Context Dependent Predictor
Nonlinear DPCM Encoding
Variable Length Block Encoding (Huffman Codes)
Error Correction Encoding
Storable Transmissible Compressed Data

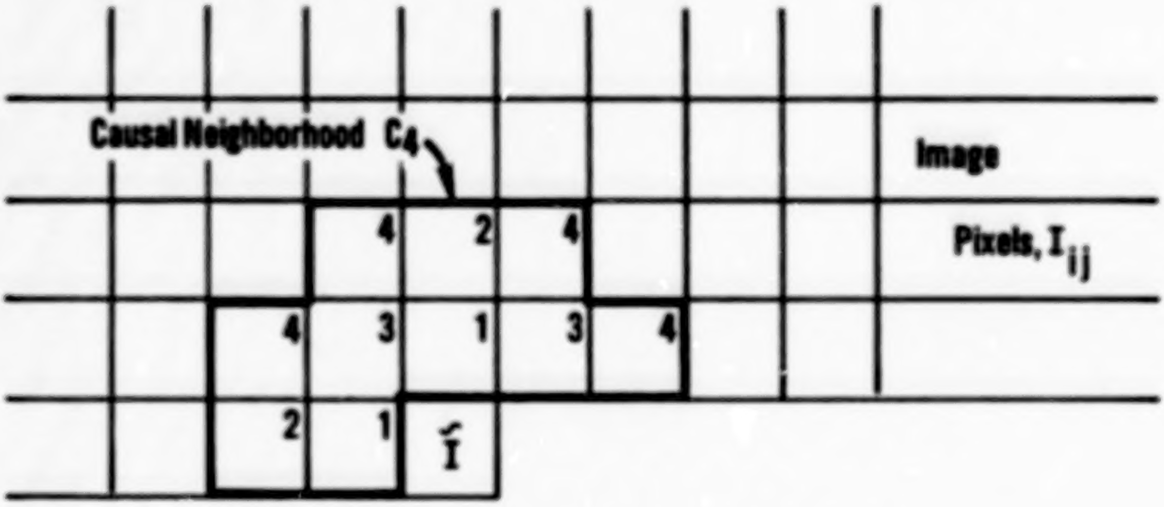
Figure 1. Data Compression Block Diagram



$$\sigma_I^2 = \underbrace{\sigma_0^2}_{\text{Electronic Noise}} + \underbrace{\Gamma I}_{\text{Shot Noise}} + \underbrace{\beta^2 I^2}_{\text{Scene Noise}}$$

Note: Output Noise is Independent of Input

Figure 2. Equal Variance Transfer Function


Causal Neighborhood C4
Image
Pixels, I ij
4
2
4
4
3
1
3
4
2
1
Î

- Context Index "K" is Determined By Image Gradient
- Context Dependent Predictor

$$I = \sum_{i,j \in C_4} \alpha^{(K)}_{i,j} \; I_{ij}$$

Figure 3. Context Dependent Bandwidth Compression

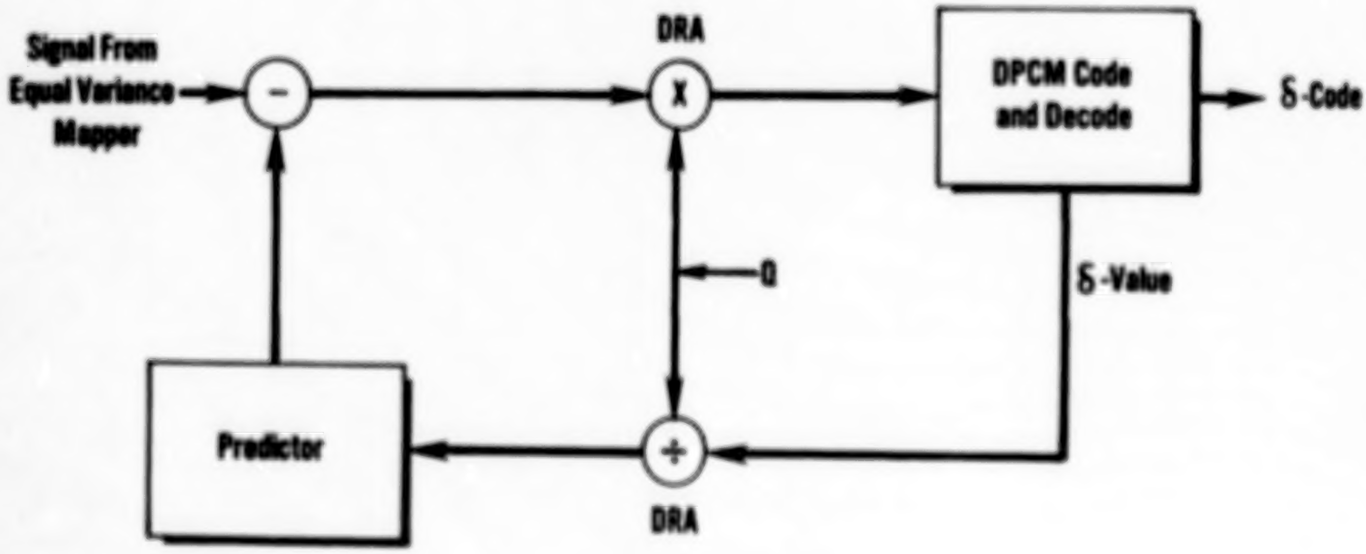

Signal From
Equal Variance
Mapper
−
DRA
X
DPCM Code
and Decode
δ-Code
Q
δ-Value
Predictor
+
DRA

Figure 4. Block Diagram of DPCM Processor

Table 1. 5-Bit DPCM Table

δ-Code State	δ_x-Range	No. of x-States	δ Value
0	-3 – -3	7	0
±1	±(4 – 9)	6	±6
±2	10 – 17	7	13
±3	18 – 28	10	21
±4	29 – 42	14	33
±5	43 – 62	20	50
±6	63 – 91	29	73
±7	92 – 132	41	106
±8	133 – 192	60	154
±9	193 – 278	86	223
±10	279 – 402	124	323
±11	403 – 582	180	467
±12	583 – 843	261	676
±13	844 – 1220	377	978
±14	1221 – 1765	545	1415
±15	1766 – ∞	–	2047

Blocking Geometry:

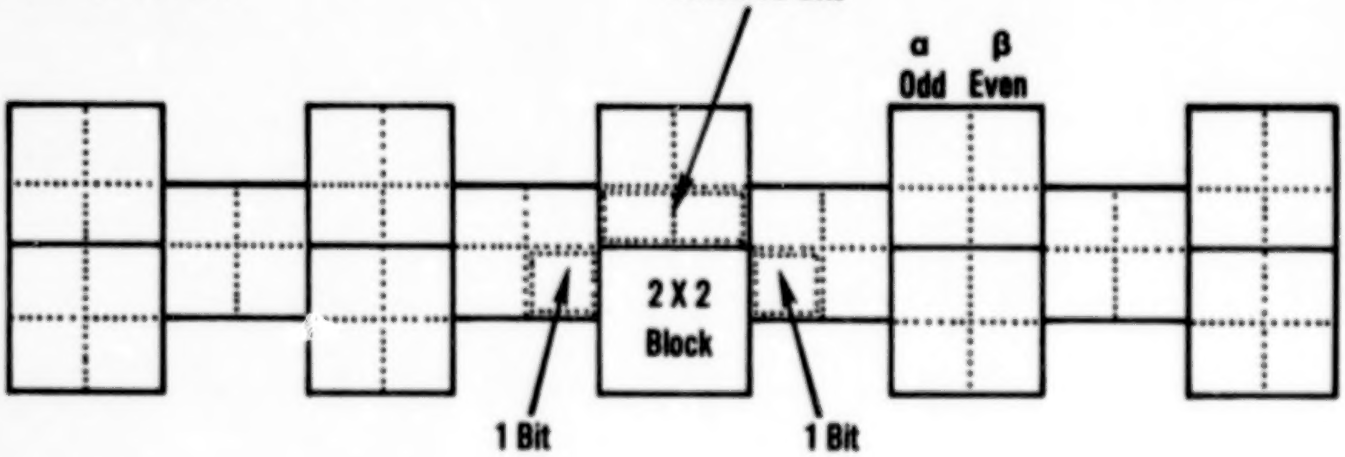

Context 2-Bits
α Odd
β Even
2 X 2 Block
1 Bit
1 Bit

Figure 5. Variable Length Encoder 2 X 2 Blocking Concept

SUB-CATEGORIES

- **8 Sub-Categories Per Pixel**
  - **Unique Categories**

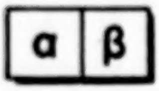
δ-Code	Category Index	Resolution Required
±0	±0	None
±1	±1	None
±2	±2	None

  - **Non-Unique Categories**

δ-Code	Category Index	Resolution Required
±(3, 4, 5, 6, 7)	±3	1 of 5
(±8 I 9 I 10....±15)	-0	1 of 16

1 X 2 CATEGORY CODES ★

β \ α	-3	-2	-1	0	1	2	3	-0
-3	27	25	25	25	25	25	27	29
-2	25	24	20	21	22	23	26	28
-1	25	19	4	5	6	9	26	28
0	25	18	3	0	7	10	26	28
1	25	17	2	1	8	11	26	28
2	25	16	15	14	13	12	26	28
3	27	26	26	26	26	26	27	29
-0	30	28	28	28	28	28	30	31



★ MS Two Bits Are Stored for Context

Figure 6. Category and Resolution Concepts

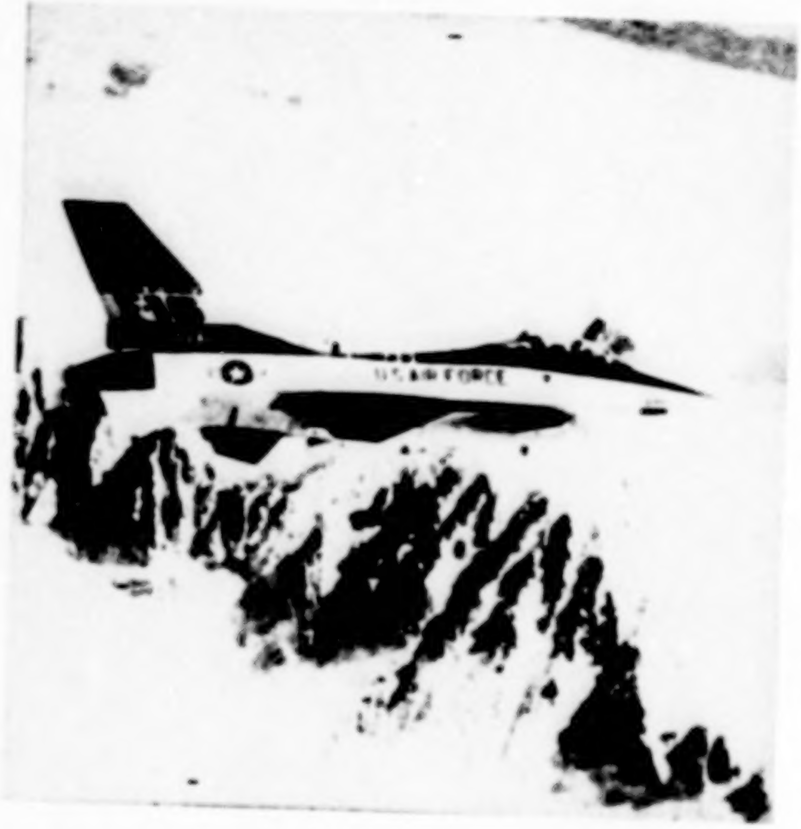


Figure 7a. Plane Original.

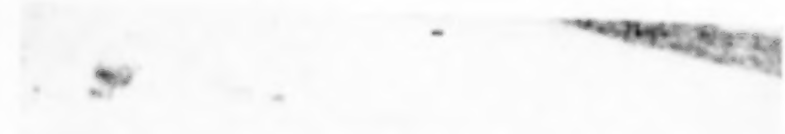


Figure 7b. Plane at 2.1 BPP.

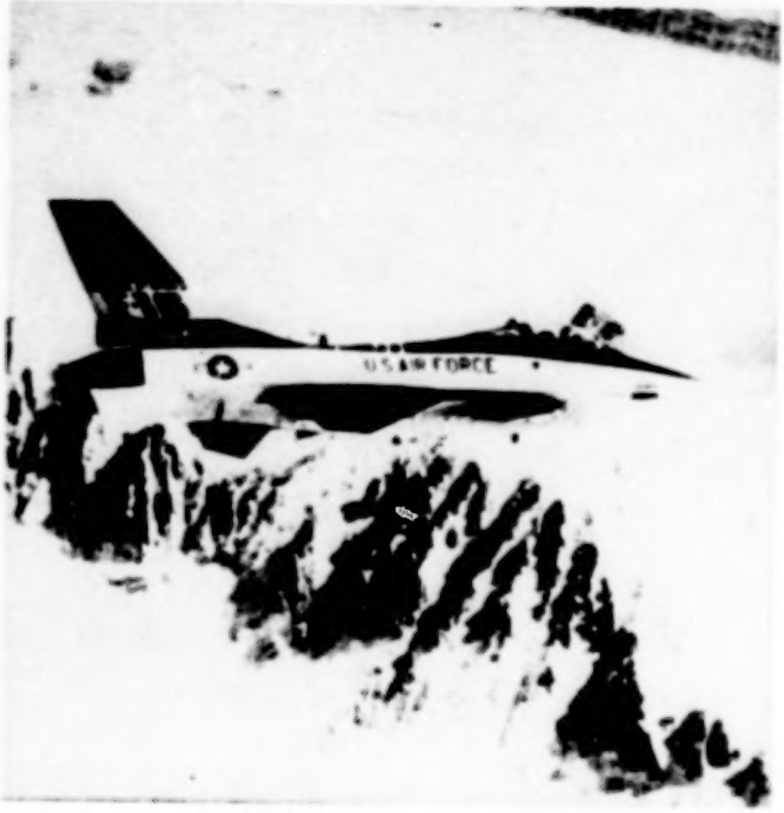

U.S. AIR FORCE

Figure 7c. Plane at 1.7 BPP.

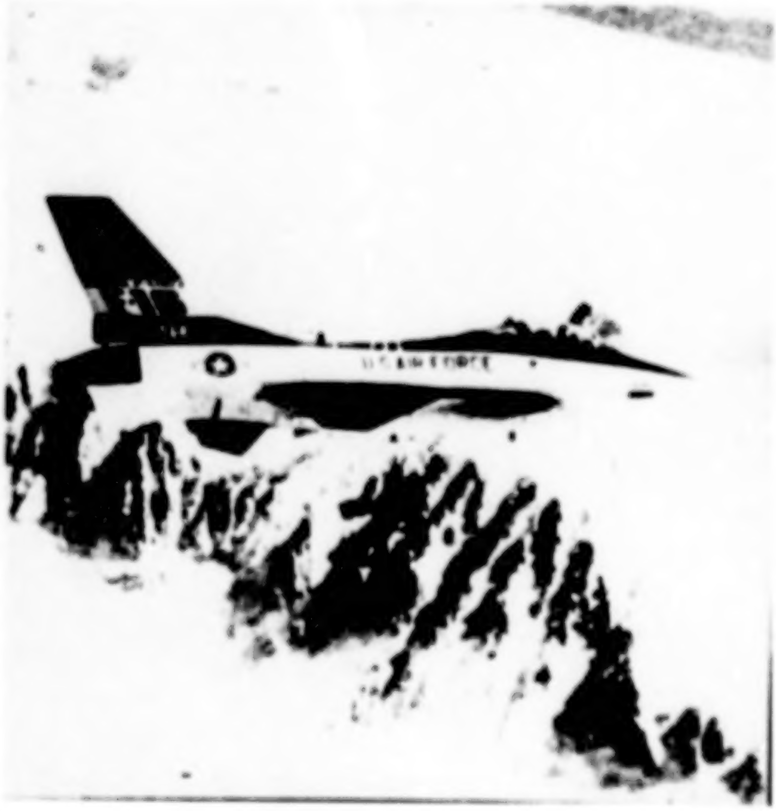

U.S. AIR FORCE

Figure 7d. Plane at 1.4 BPP.

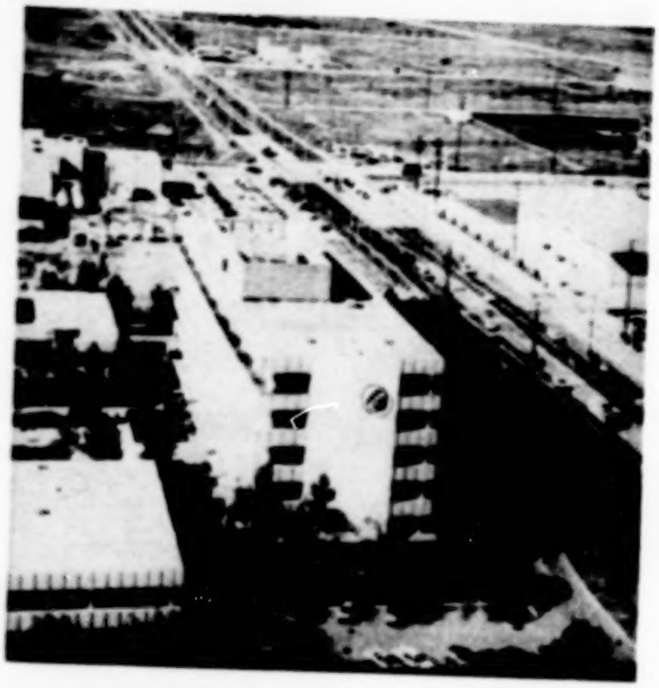


Figure 8a. Building Original.

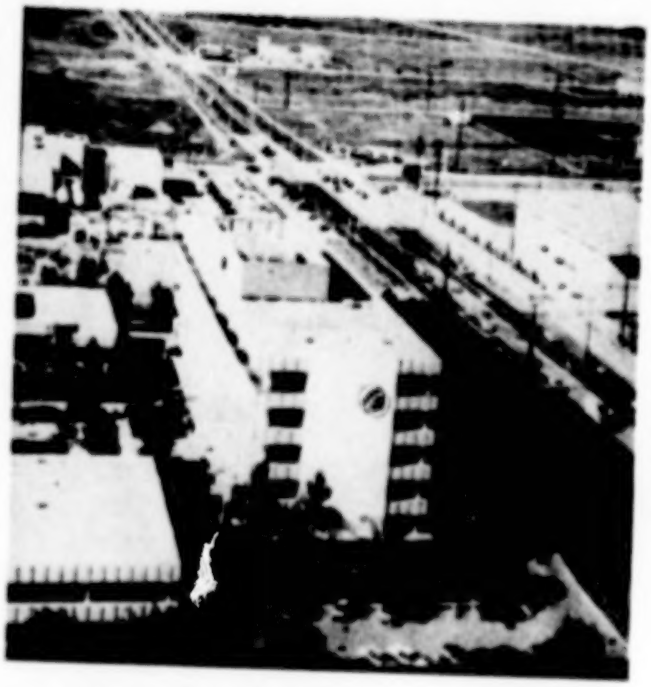


Figure 8b. Building at 2.4 BPP.

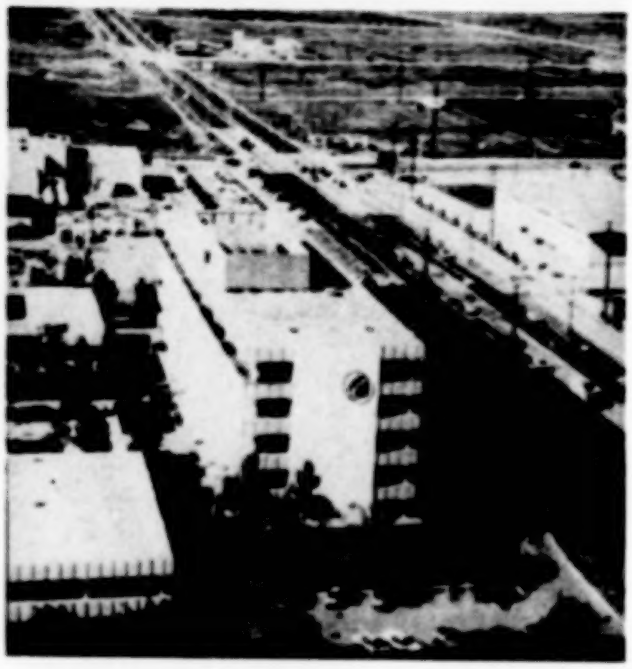


Figure 8c. Building at 2.0 BPP.

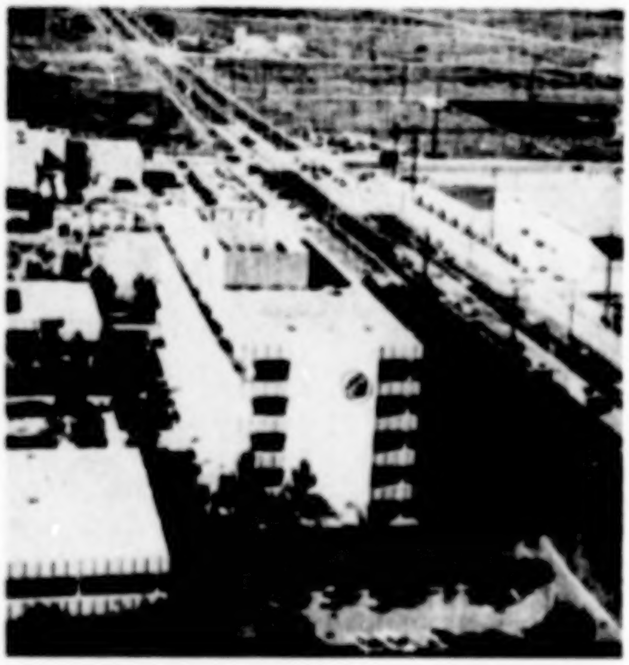


Figure 8d. Building at 1.5 BPP.

IMAGE GATHERING AND CODING FOR DIGITAL RESTORATION: INFORMATION EFFICIENCY AND VISUAL QUALITY

Friedrich O. Huck
NASA Langley Research Center, Hampton, Virginia

Sarah John, Judith A. McCormick, and
Ramkumar Narayanswamy
Science and Technology Corporation, Hampton, Virginia

ABSTRACT

Image gathering and coding are commonly treated as tasks separate from each other and from the digital processing used to restore and enhance the images. Our goal in this paper is to develop a method that allows us to assess quantitatively the combined performance of image gathering and coding for the digital restoration of images with high visual quality. Digital restoration is often interactive because visual quality depends on perceptual rather than mathematical considerations, and these considerations vary with the target, the application, and the observer. Our approach is based on the theoretical treatment of image gathering as a communication channel [J. Opt. Soc. Am. A2, 1644 (1985); 5,285 (1988)]. Initial results suggest that the practical upper limit of the information contained in the acquired image data ranges typically from ~ 2 to 4 binary information units (bifs) per sample, depending on the design of the image-gathering system. The associated information efficiency of the transmitted data (i.e., the ratio of information over data) ranges typically from ~ 0.3 to 0.5 bif per bit without coding to ~ 0.5 to 0.9 bif per bit with lossless predictive compression and Huffman coding. These upper limits of performance are reached when the sampling passband of the image-gathering system closely matches the Wiener spectrum of the incident radiance field. The visual quality that can be attained with interactive image restoration improves perceptibly as the available information increases to ~ 3 bifs per sample. However, the perceptual improvements that can be attained with further increases in information are very subtle and depend on the target and the desired enhancement.

1. INTRODUCTION

Image gathering and coding are commonly treated as tasks separate from each other and from the digital processing used to restore and enhance the images. Ordinarily, image-gathering systems are designed to produce good visual quality for conventional image displays, and data-compression techniques are developed to reduce, as much as possible, the data necessary to reproduce a faithful duplicate of this original image. Image restoration and enhancement, despite the rapidly increasing use of digital processing, are virtually ignored in these assessments of image gathering and coding.

Digital image restoration is often interactive because a single figure of merit for visual quality does not exist to formulate a single "best" algorithm. Visual quality is too elusive a concept for such a figure to exist. It depends on a number of attributes, such as fidelity (resemblance to the scene), resolution (minimum discernible detail), sharpness (contrast between large areas), and clarity (absence of visual artifacts and noise). The trade-off between these attributes of image quality still must be based on perceptual rather than mathematical considerations, and these considerations vary with the target, the application, and the observer. In addition, sometimes the enhancement of certain target features is desirable to improve resolution and sharpness, even at the cost of fidelity and clarity.

In previous papers Huck et al.[1,2] have shown that image gathering can be treated like a communication channel if (and only if) the image-gathering degradations are correctly accounted for in image processing. If this is done, then the informationally optimized image-gathering system tends to maximize the fidelity and robustness of a variety of optimally restored representations ranging from images to edges. It also is possible with interactive image restoration to improve significantly on the visual quality produced by the traditional methods employed in digital image gathering and restoration.[3,4] These traditional methods[5-10] often have failed to improve on the visual quality obtained in a simpler and faster way by image reconstruction and interpolation. It is perhaps for this reason, at least in part, that image gathering and coding have not been assessed directly with digital restoration and enhancement in the past.

In this paper we extend the information theoretic assessment of image gathering to include image coding. Our goal is to develop a method that allows us to assess quantitatively the combined performance of image gathering and coding for the interactive restoration of images with high visual quality. The major questions we deal with are: How much visual information can be acquired by the image-gathering process? How much information is required to restore images with high visual quality? And how can this information be transmitted most efficiently? We do not attempt here to compare the performance of a variety of image-coding techniques. Instead, we limit our assessment to the familiar lossless predictive compression with Huffman coding.

2. OUTLINE

Figure 1 presents the end-to-end block diagram of the image gathering, coding, and restoration processes that we analyze in this paper. Our approach is to assess quantitatively the flow of information through the image gathering and coding processes, followed by a qualitative assessment of the visual quality that can be restored from the transmitted information.

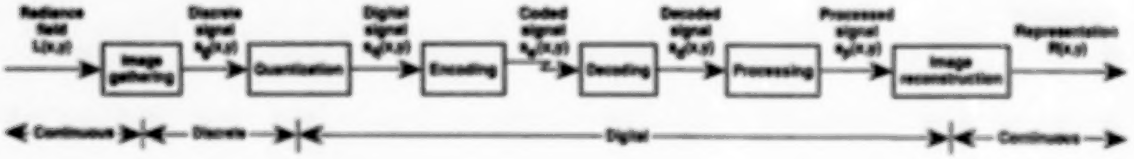


Figure 1. Model of image gathering, coding, and restoration.

In Section 3 we assess the information density of the data that is acquired by the image-gathering process in terms of the Wiener spectrum of the radiance field, the design of the image- gathering system, and the dynamic range and quantization intervals of the quantizer. We also introduce the concept of information efficiency (i.e., the ratio of information over data) as an additional criterion of the effectiveness of image gathering and coding. Our formulations are based on the theoretical treatment of image gathering given by Huck et al.[1,2] Following the methods of Shannon[11] and of Fellgett and Linfoot,[12] this treatment is constrained by the assumption that the radiance field and the noise are wide-sense-stationary Gaussian random processes.

In Section 4 we assess the effects of image coding on the information efficiency of the transmitted data. We use the familiar lossless predictive compression together with Huffman coding. The predictive compression reduces the statistical redundancy of the digital data without loss of information, and the Huffman coding compresses the data by transmitting the more probable symbols in fewer bits than the less probable ones. In addition, we demonstrate that important differences exist between the information density of the transmitted data and the entropy that is often used in the prevailing digital processing literature[5-10] to assess data compression.

In Section 5 we assess the quality of the restored images as a function of the available information. We first consider fidelity-maximized restorations. These restorations allow us to perform parametric trade-offs in terms of a single figure of merit, namely, the image fidelity. However, the fidelity-maximized images exhibit some visual defects such as ringing, aliasing artifacts, and noise. Thus, we use the Wiener-Gaussian enhancement (WIGE) filter introduced by McCormick et al.[4] to suppress these defects and improve the visual quality.

3. IMAGE GATHERING

A. Information Capacity

Let the image-gathering system acquire information about some isoplanatism area A of the radiance field $L(x,y)$ with the average power σ_L^2. Furthermore, let the image-gathering process be constrained, like a communication channel, only by the frequency passband $\widehat{B}$ and the white noise $n(x,y)$ with the power σ_N^2. Then the absolute upper limit of the acquired information about the area A is defined by the expression

$$H_c = \frac{1}{2}|A||\widehat{B}|\log_2\left[1 + (K\sigma_L/\sigma_N)^2\right],$$

where $K\sigma_L/\sigma_N$ is the rms signal-to-noise ratio (SNR) and K is the steady-state gain of the radiance-to-signal conversion in the image-gathering process. The associated information capacity of the image-gathering process per unit area A and unit passband $\widehat{B}$ is

$$h_c = \frac{H_c}{|A||\widehat{B}|} = \frac{1}{2}\log_2\left[1 + (K\sigma_L/\sigma_N)^2\right]. \tag{1}$$

The magnitude of h_c may be defined as bifs, binary information units per unit area and passband, analogous to bits for the binary units per sample of the transmitted digital data.

The information capacity h_c is plotted as a function of the SNR $K\sigma_L/\sigma_N$ in Fig. 2. It varies from ~ 1 bif for $K\sigma_L/\sigma_N = 2$ to ~ 10 bifs for $K\sigma_L/\sigma_N = 1000$. This is the range of SNR's that is typically of interest. SNR's below this range ordinarily do not permit the restoration of images with good visual quality, and SNR's above this range ordinarily do not improve the visual quality. In practice, of course, information is inevitably lost because the Wiener spectrum of random radiance fields is not white and band-limited to $\widehat{B}$ and because the image-gathering process introduces aliasing and blurring as well as noise. In addition, information ordinarily is lost by the signal quantization that is required for digital data transmission and processing.

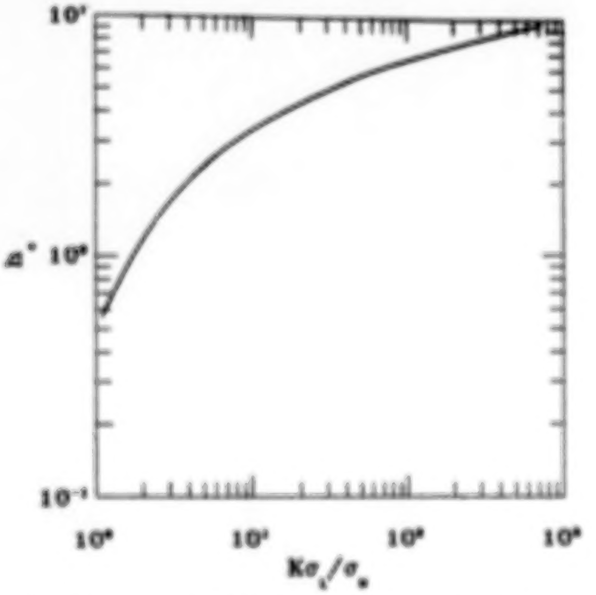


Figure 2. Information capacity h_c versus SNR $K\sigma_L/\sigma_N$. The image-gathering process is assumed to be constrained, like a communication channel, only by the sampling passband $\widehat{B}$ and the SNR $K\sigma_L/\sigma_N$. The radiance field spectrum is assumed to be white and band limited to $\widehat{B}$.

B. Radiance-Field Properties

We assume that the incident radiance field $L(x,y)$ consists of contiguous rectangles whose sides are parallel to some axes (x', y') (see Fig. 3). The transitions along each axis obey the Poisson probability-density function with the (expected) mean separation λ^{-1}, and the radiance-field magnitude of each rectangle obeys the zero-mean Gaussian probability-density function with the (expected) variance σ_L^2. The resultant autocorrelation of $L(x,y)$ is[13]

$$\begin{aligned}\Phi_L(x,y) &= \sigma_L^2 \exp\left[-\lambda(1-c)\left(|x'|+|y'|\right)\right]\\ &= \sigma_L^2 \exp\left[-\left(|x'|+|y'|\right)/\mu\right]\end{aligned}$$

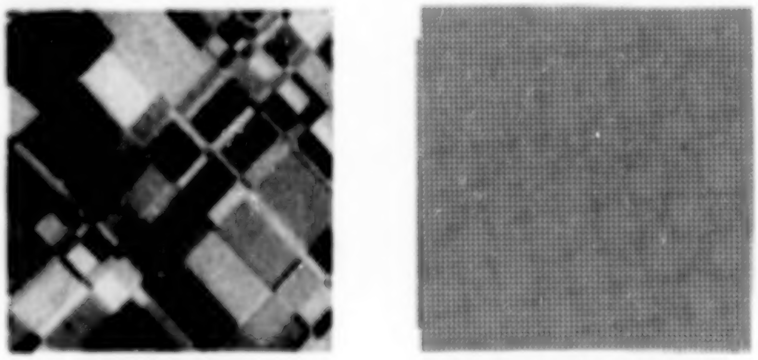
where c is the correlation of the radiance-field magnitudes of adjacent rectangles and, for convenience, $\mu = 1/\lambda(1-c)$. If we let the orientation of the (x', y') axes of the rectangles be random with uniform probability, then the autocorrelation becomes circularly symmetric as given by

$$\Phi_L(x,y) = \Phi_L(r) = \sigma_L^2 \exp\left(-|r|/\mu\right), \tag{2a}$$

where $r^2 = x^2 + y^2$. The corresponding Wiener spectrum of $L(x,y)$ can be closely approximated by[1-4,14,15]

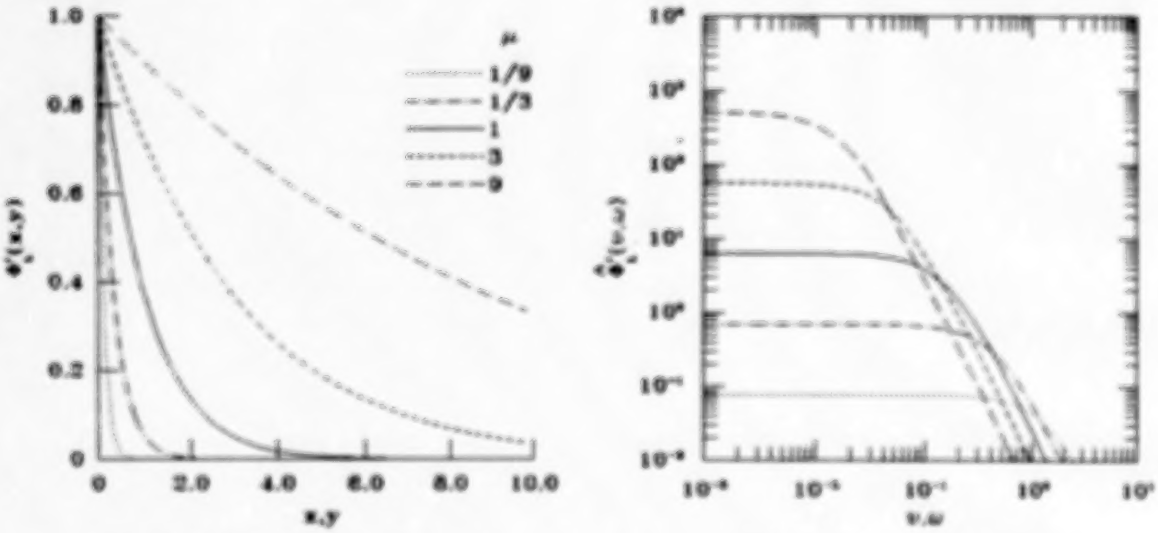
$$\hat{\Phi}_L(v,w) = \hat{\Phi}_L(\rho) = \frac{2\pi\mu^2\sigma_L^2}{\left[1+(2\pi\mu\rho)^2\right]^{3/2}}, \tag{2b}$$

where $\rho^2 = v^2 + w^2$. Figure 4 illustrates the normalized auto correlation $\Phi'_L(x,y) = \sigma_L^{-2}\Phi_L(x,y)$ and the normalized Wiener spectrum $\hat{\Phi}'_L(v,w) = \sigma_L^{-2}\Phi_L(v,w)$. The mean spatial detail of the radiance field is conveniently represented by μ. This implies that the correlation is $c \approx 0.3$. The exact expression for the Wiener spectrum of the target shown in Fig. 3 is given by Fales et al.[3] [Eq. (18)]. The Wiener-spectrum curves for that expression are almost identical to those given for the more convenient Eq. (2b).



(a) Target (b) Sampling lattice

Figure 3. Random radiance field with mean spatial detail $\mu = 3$, and the sampling lattice with unit sampling intervals.



(a) Autocorrelation

(b) Wiener spectrum

Figure 4. Autocorrelation functions and Wiener spectra of the radiance field for several mean spatial details μ.

C. Image-Gathering Degradations

Conventional image-gathering systems consist of an objective lens (or lens system) and some sort of photon-detection and sampling mechanism. The most common mechanisms are sensor-array and line-scan devices. The lens and photosensor apertures are basically low-pass spatial-frequency filters. The spatial-frequency response of the image-gathering system, which is the product of these two low-pass-filter responses, ordinarily decreases smoothly with increasing spatial frequency until the lens diffraction limit is reached.

Figure 5 presents a model of the image-gathering process that transforms the continuous radiance field $L(x,y)$ into the signal $s_g(x,y)$ as defined by the expression

$$s_g(x,y) = \left[K L(x,y) * \tau_g(x,y)\right] \text{III}(x,y) + n(x,y), \tag{3a}$$

where K is the steady-state gain of the (linear) radiance-to-signal conversion, $n(x,y)$ is the (additive, discrete) sensor noise, $*$ denotes convolution, and $\text{III}(x,y)$ denotes sampling. Taking the Fourier transform of $s_g(x,y)$ yields the spatial-frequency representation of the acquired signal

$$\hat{s}_g(v,w) = K\hat{L}(v,w)\hat{\tau}_g(v,w) * \widehat{\text{III}}(v,w) + \hat{n}(v,w), \tag{3b}$$

where $\hat{L}(v,w)$ and $\hat{n}(v,w)$ are the spatial-radiance and noise transforms, respectively, and $\hat{\tau}_g(v,w)$ is the spatial-frequency response of the image-gathering system. The sampling function is given by

$$\begin{aligned}\widehat{\text{III}}(v,w) &= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(v-m, w-n) \\ &= \delta(v,w) + \widehat{\text{III}}_{\neq 0,0}(v,w)\end{aligned}$$

for unit sampling intervals. The term $\widehat{\text{III}}_{\neq 0,0}(v,w)$ accounts for the sampling sidebands.

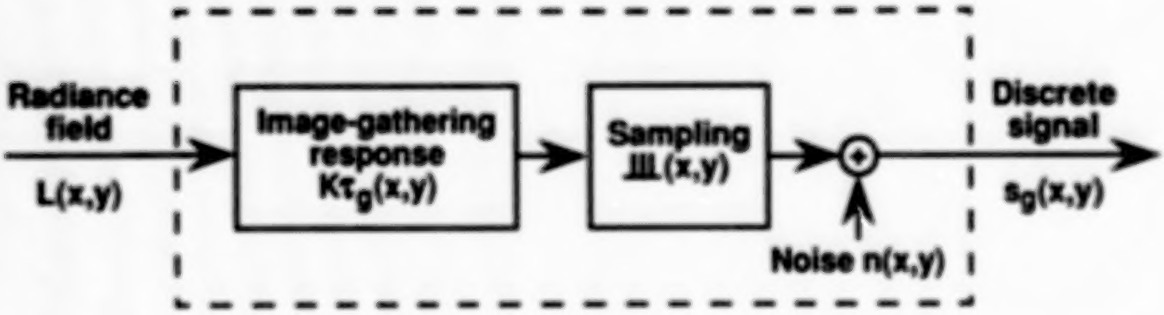


Figure 5. Model of the image-gathering process.

It is convenient to normalize the sampling intervals to unity, and to define the frequency passband $\hat{B}$ as the sampling passband given by

$$\hat{B} = \{(v,w), |v| < 0.5,\ |w| < 0.5\}. \tag{4}$$

The corresponding area in the frequency domain is $|\hat{B}| = 1$. The low-pass frequency response of conventional image-gathering systems can often be approximated by the Gaussian form[16]

$$\hat{\tau}_g(v,w) = \exp\left[-(\rho/\rho_c)^2\right], \tag{5}$$

where the optical-design parameter ρ_c is the spatial frequency at which $\hat{\tau}_g(v,w) = 1/e = 0.37$.

If we now let the image-gathering process be constrained by the response $\hat{\tau}_g(v,w)$, the sampling passband $\hat{B}$, and the SNR $K\sigma_L/\sigma_N$, then the information density h_g of the acquired signal $s_g(x,y)$ becomes[1]

$$h_g = \frac{1}{2}\iint_{\hat{B}} \log_2\left[1 + \frac{\hat{\Phi}'_L(v,w)|\hat{\tau}_g(v,w)|^2}{\hat{\Phi}'_L(v,w)|\hat{\tau}(v,w)|^2 * \hat{\text{Ш}}_{\neq 0,0}(v,w) + (K\sigma_L/\sigma_N)^{-2}}\right] dv\, dw. \tag{6}$$

The associated variance σ_g^2 of the signal is

$$\sigma_g^2 = \sigma_L^2 \iint_{-\infty}^{\infty} \hat{\Phi}'_L(v,w)|\hat{\tau}_g(v,w)|^2\, dv\, dw. \tag{7}$$

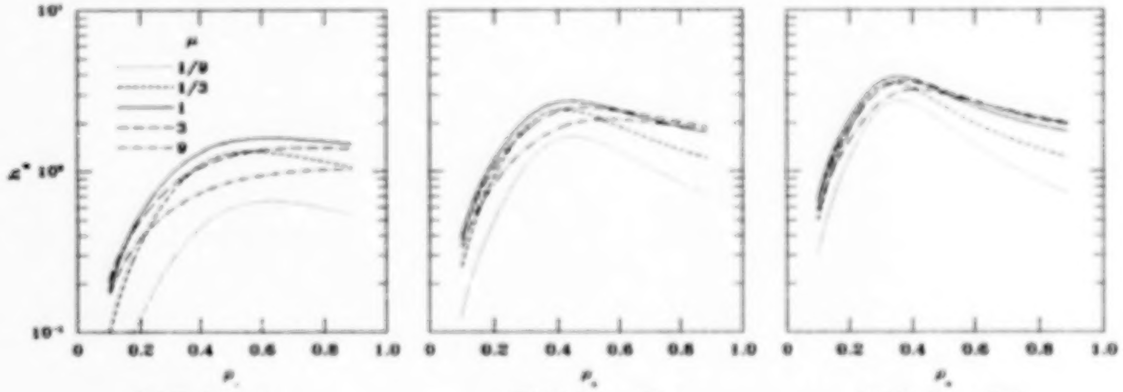
Figures 6 and 7 illustrate the dependence of the information density h_g on the optical design parameter ρ_c for several mean spatial details μ and SNR's $K\sigma_L/\sigma_N$. These results suggest the two following generalizations:

(1) The information density h_g tends to be maximum when the sampling passband most closely matches the radiance-field spectrum, regardless of the design of the image-gathering system. This occurs, for the target characterized by Eqs. (2), when the sampling intervals are approximately equal to the mean spatial detail (i.e., when $\mu \approx 1$).

This generalization intuitively is appealing when one considers image restoration. One could not expect to restore spatial detail that is much finer than the sampling interval, and one ordinarily could expect to restore detail that is much coarser from fewer samples.

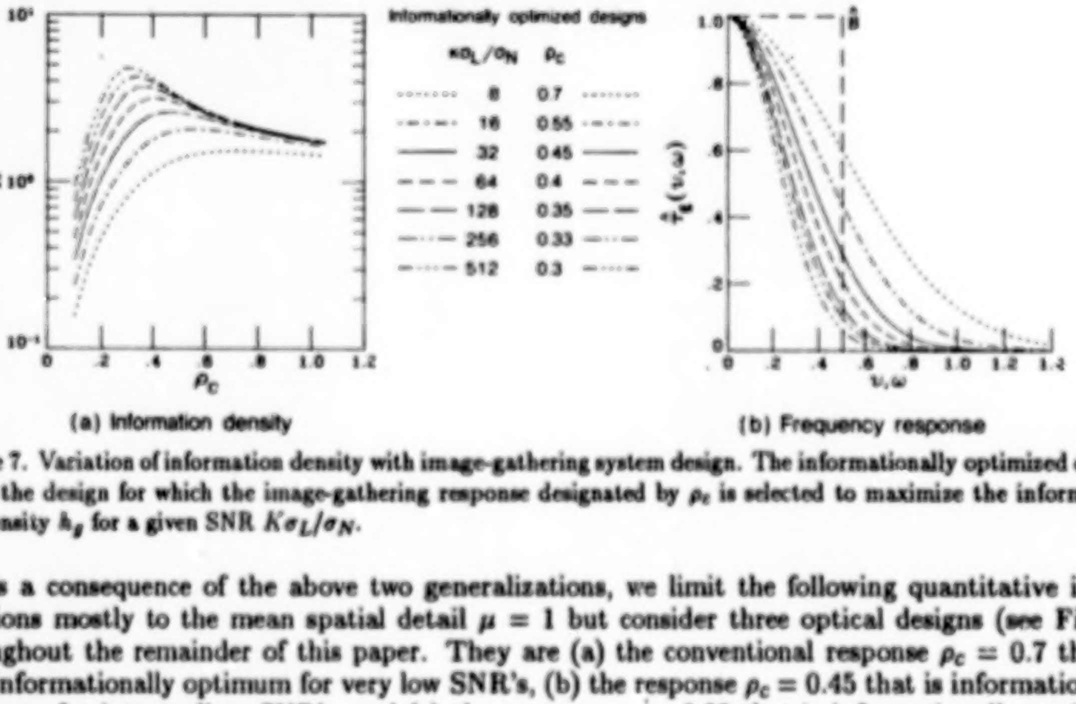
(2) The informationally optimized optical design (i.e., trade-off between aliasing and blurring) is a function of the SNR.

Again, this generalization intuitively is appealing when one considers image restoration. In one extreme, when the SNR is very low, then the restoration of fine detail is constrained by noise, and so it ordinarily would be preferable to avoid substantial blurring (at the cost of aliasing). In the other extreme, when the SNR is very high, then the restoration of fine detail is not constrained by noise, and so it ordinarily would be preferable to avoid substantial aliasing (at the cost of blurring). However, as we will show in Section 5 below, some other constraints may be introduced by the image restoration, such as ringing near sharp edges (Gibb's phenomenon).



(a) $K\sigma_L/\sigma_N = 8$ (b) $K\sigma_L/\sigma_N = 32$ (c) $K\sigma_L/\sigma_N = 128$

Figure 6. Variation of information density with image-gathering system design. Results are given for several mean spatial details μ and SNR's $K\sigma_L/\sigma_N$.



(a) Information density (b) Frequency response

Figure 7. Variation of information density with image-gathering system design. The informationally optimized design is the design for which the image-gathering response designated by ρ_c is selected to maximize the information density h_g for a given SNR $K\sigma_L/\sigma_N$.

As a consequence of the above two generalizations, we limit the following quantitative investigations mostly to the mean spatial detail $\mu = 1$ but consider three optical designs (see Fig. 7) throughout the remainder of this paper. They are (a) the conventional response $\rho_c = 0.7$ that is also informationally optimum for very low SNR's, (b) the response $\rho_c = 0.45$ that is informationally optimum for intermediate SNR's, and (c) the response $\rho_c = 0.35$ that is informationally optimum for very high SNR's.

D. Quantization

Each discrete signal $s_g(x, y)$ is quantized into κ levels for η-bit encoding, $\kappa = 2^\eta$. If we divide the area A into M by N samples, then the area of A for unit sampling intervals is $|A| = MN$. Thus, the number of distinguishable states in A is κ^{MN}, and the amount of data in A is

$$H_d = MN \log_2 \kappa.$$

The associated data density is

$$h_d = \frac{H_d}{MN} \log_2 \kappa = \eta. \tag{8}$$

It is convenient to let the units of h_d be bits even though strictly they are bits per sample. Just as the information capacity h_c given by Eq. (1) sets a theoretical upper limit on the information density h_g acquired by image gathering, so the data density h_d given by Eq. (8) sets a theoretical upper limit on the information density transmitted by digital communication.

E. Information Efficiency

Since it ordinarily is desirable to use as few encoding levels as possible, some loss of information density is associated with the quantization process. Hence, the information density h_q of the quantized signal $s_q(x, y)$ is closely interrelated with the data density h_d. This interrelationship suggests the definition of information efficiency as the ratio h_q/h_d. The units of this ratio are bif/bit. This definition of information efficiency is analogous to Khinchin's definition of "relative entropy" as the ratio $h/\log\, m$, where h is the entropy of the test, and $\log m$ is the maximum value of h for the m different symbols of the test.[17] Another analogy is Jones's definition of "information efficiency" of a light beam as the information capacity per transmitted photon.[18]

To properly interpret the information efficiency h_q/h_d, we must account for an important difference between continuous and discrete entropies. The data density h_d is defined for a *discrete* random variable (i.e., the quantization levels with a uniform probability density function) for which the entropy provides i an absolute measure of randomness. The information density h_q is defined for a *continuous* random variable (i.e., the continuous magnitude with a Gaussian probability density function) for which the entropy provides a measure of randomness relative to an assumed standard. It intuitively is satisfying to adjust the ratio h_q/h_d so that the theoretical upper limit of information efficiency becomes unity. This adjustment occurs when (1) the Wiener spectrum $\widehat{\Phi}_L(v, w)$ is white and band-limited to $\widehat{B}$, (2) the image-gathering response $\hat{\tau}_g(v, w)$ is unity within $\widehat{B}$ and zero outside, and (3) the quantization intervals are very large compared with the magnitude of the noise. The information density h_q of the digital data becomes then

$$h_q = \frac{1}{2} \iint_{\widehat{B}} \log_2 \left[1 + \frac{\widehat{\Phi}'_L(v, w) |\hat{\tau}_g(v, w)|^2}{\widehat{\Phi}'_L(v, w) |\hat{\tau}_g(v, w)^2 * \widehat{\text{Ш}}_{\neq 0,0}(v, w) + (K\sigma_L/\sigma_N)^{-2} + \kappa^{-2}} \right] dv\, dw. \tag{9}$$

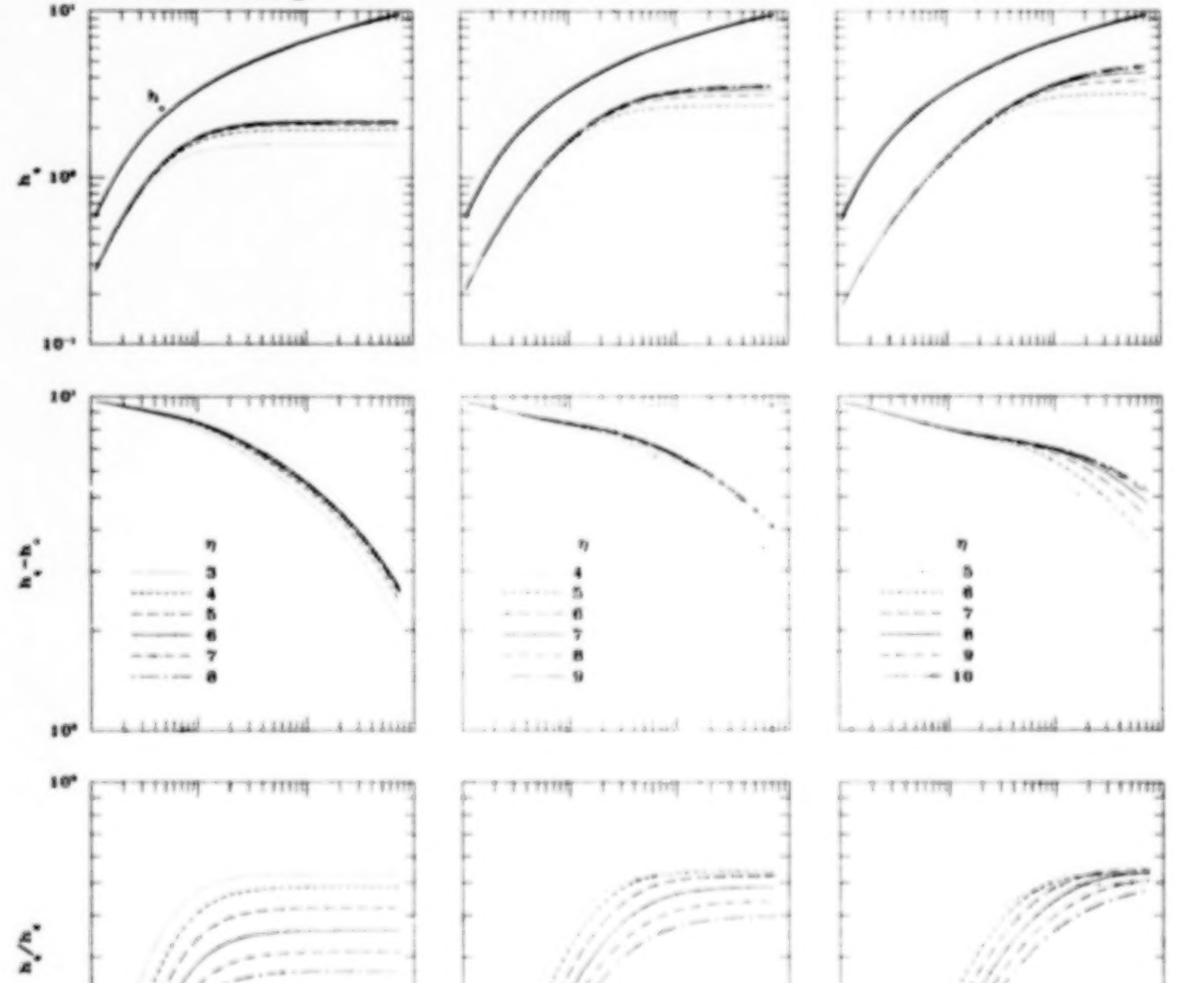
Equation (9) for the information density h_q reduces to Eq. (8) for the data density h_d when the above three conditions are evoked. The final step in this reduction of Eq. (9) to Eq. (8) entails the approximation given by

$$h_q = \frac{1}{2} \log_2(1 + \kappa^2) \approx \log_2 \kappa = h_d.$$

This adjustment of the information density h_q leads to a linear encoding of the Gaussian signal variation over a dynamic range of $-\sqrt{3}K\sigma_g$ to $\sqrt{3}K\sigma_g$, which encompasses 92% of the signal. The corresponding quantization interval is $\Delta = 2\sqrt{3}K\sigma_g/\kappa$. Values of $s_g(x,y) < \bar{s}_g - \sqrt{3}\sigma_g$ are assigned the value 0 and values of $s_g(x,y) > \bar{s}_g + \sqrt{3}\sigma_g$ are assigned the value $\kappa - 1$, where $\bar{s}_g$ is the average value of $s_g(x,y)$.

Figures 8 and 9 illustrate the dependence of the information density h_q and the information efficiency h_q/h_d on the optical-design parameter ρ_c, the SNR $K\sigma_L/\sigma_N$, and the number of encoding levels η. These results suggest the following generalizations:

(3a) Conventional optical responses ($\rho_c = 0.7$) limit the information density to $h_q \approx 2.2$ bifs. This limit is closely approached when the SNR is $K\sigma_L/\sigma_N \approx 20$ and the number of encoding levels is $\eta \approx 6$ bits. The corresponding maximum information efficiency is $h_q/h_d \approx 0.53$ bif/bit, but with the reduced information density $h_q \approx 1.7$ bifs is obtained with 3-bit encoding.



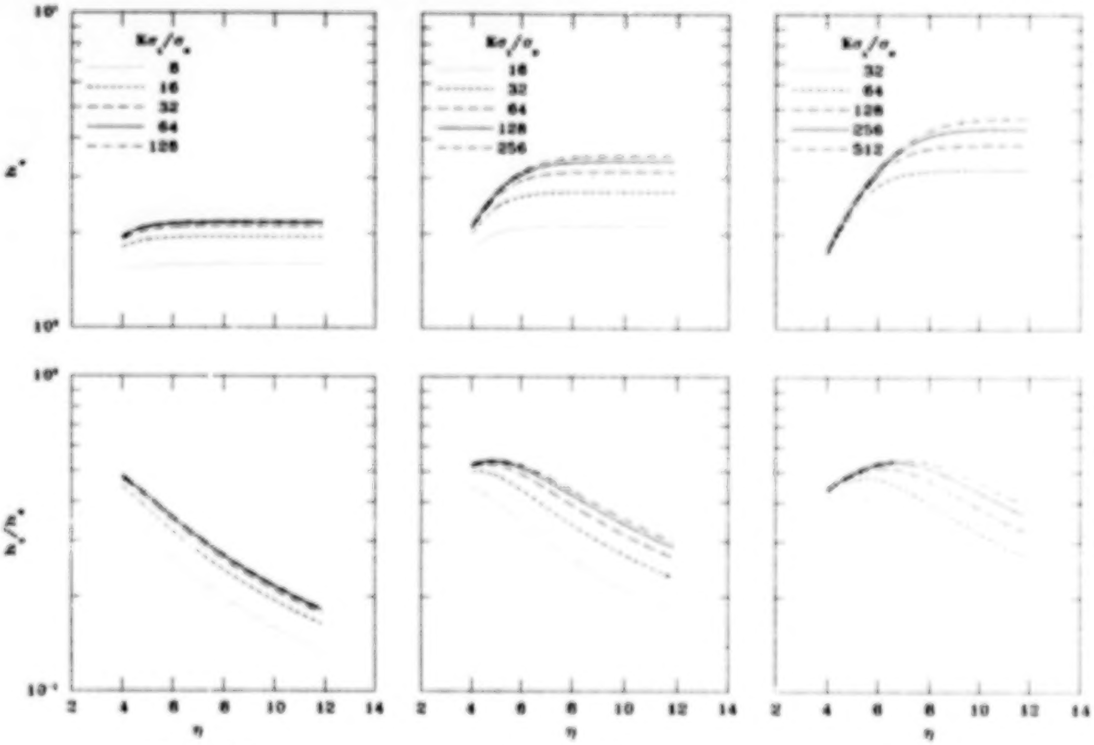
(a) $\rho_c = 0.7$ (b) $\rho_c = 0.45$ (c) $\rho_c = 0.35$

Figure 8. Information versus the SNR $K\sigma_L/\sigma_N$ for several encoding levels η. The first row presents the information capacity h_c and density h_q, the second row presents the lost information $h_q - h_c$, and the third row presents the information efficiency h_q/h_d. The image-gathering system is characterized by the optical-design parameter ρ_c and the SNR $K\sigma_L/\sigma_N$. The mean spatial detail $\mu = 1$.

(3b)Optical responses ($\rho_c = 0.45$) that are informationally optimized for intermediate SNR's limit the information density to $h_q \approx 3.6$ bifs. This limit is closely approached when the SNR is $K\sigma_L/\sigma_N$ 80 and the number of encoding levels is $\eta \approx 7$. The corresponding maximum information efficiency is $h_q/h_d \approx 0.54$ bif/bit, but with the reduced information density $h_q \approx 2.7$ bifs as obtained with 5-bit encoding.

(3c)Optical responses ($\rho_c = 0.35$) that are informationally optimized for high SNR's limit the information density to $h_q \approx 4.7$ bifs. This limit is closely approached when the SNR is $K\sigma_L/\sigma_N \approx 240$ and the number of encoding levels is $\eta \approx 8$. The corresponding maximum information efficiency is $h_q/h_d \approx 0.55$ bif/bit, but with the reduced information density $hq \approx 3.7$ bifs as obtained with 7-bit encoding.

The preferred number of encoding levels for information density, ranging from $\eta \approx 6$ for low SNR's to $\eta \approx 8$ for high SNR's, corresponds closely to those often encountered in practice. However, their selection entails some conflict between information density and efficiency. This conflict resolves itself with data compression. Furthermore, as these results foreshadow, it is the image-gathering system that is designed for highest information density that also provides the highest information efficiency with lossless image coding.



(a) ρ_c = 0.7

(b) ρ_c = 0.45

(c) ρ_c = 0.35

Figure 9. Information density h_q and efficiency h_q/h_d versus the encoding level η for several SNR's. The image-gathering system is characterized by the optical-design parameter ρ_c and the SNR $K\sigma_L/\sigma_N$. The mean spatial detail $\mu = 1$.

4. DATA COMPRESSION

A. Entropy Versus Information Density

It is common in the prevailing literature[5-10] to consider information density to be synonymous with the entropy used to assess data compression. However, entropy, unlike information density, does not distinguish between the properties of the scene that we wish to restore and the degradations of the image-gathering process that we wish to minimize. For example, whereas aliasing and noise subtract from the information density, these same degradations add to the entropy. Since it is not possible to distinguish quantitatively between the desired and undesired components of the signal, it also is not possible to measure the information density of an image. However, it is possible, at least in theory, to measure its entropy.

The entropy of the digital signal $s_q(x,y)$ can be determined as follows. Let $p(z_1, z_2, ..., z_{MN})$ be the probability of realizing a particular set of digital data containing MN samples and κ quantization levels per sample. Then the entropy θ_q of this data is defined as the logarithm of the probable number of alternate, distinguishable sets given by

$$\theta_q = -\frac{1}{MN}\sum_{z_1=1}^{\kappa}\sum_{z_2=1}^{\kappa}\cdots\sum_{z_{MN}=1}^{\kappa} p(z_1, z_2, \ldots, z_{MN})\log_2 p(z_1, z_2, \ldots, z_{MN}). \qquad (10)$$

The entropy θ_q given by Eq. (10) is equal to the information density h_g given by Eq. (9) only if the undesired components of the signal are negligible. However, since these undesired components are ordinarily not negligible, the information density h_g seldom reaches the entropy θ_q (i.e., $h_q < \theta_q$).

The amount of computation required to find θ_q given by Eq. (10) is, in practice, prohibitive. An upper boundary for the entropy θ_q can readily be found if we assume that each sample is independent of its neighbors, i.e., that

$$p(z_1, z_2, \ldots, z_{MN}) = p(z_1)p(z_2)\ldots p(z_{MN}),$$

where $z_i = \Delta k_i$ and k_i is an integer, $1 \leq k_1 \leq \kappa$ [Fig. 10(a)]. Letting $p_i \equiv p(z_i) \equiv p(\Delta k_i)$, the upper boundary for θ_q is then given by

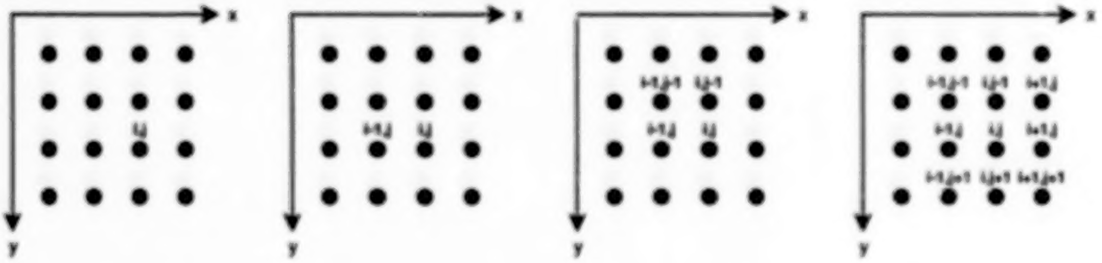
$$\theta_{q0} = -\sum_{i=1}^{\kappa} p_i \log_2 p_i. \qquad (11)$$

Ordinarily, the values p_i are obtained from the probability distribution (histogram) of the digital data $s_q(x,y)$. If we were to assume that the probability distribution is uniform so that all quantization levels are equally likely, then $p_i = 1/\kappa$ and the upper boundary θ_{q0} of the entropy θ_q would be equal to the data density h_d, i.e.,

$$\theta_{q0} = -\sum_{i=1}^{\kappa}\frac{1}{\kappa}\log_2\kappa = \eta = h_d.$$

However, neighboring samples ordinarily are not independent. As shown in Fig. 4(a), significant correlation exists out to approximately three neighboring samples (in all directions) if the mean spatial detail $\mu = 1$, and out to about 10 neighboring samples if $\mu = 3$. Blurring in the image-gathering process will further increase the correlation among neighboring samples for the fine spatial detail. In practice, it is common to consider only the nearest samples as depicted in Fig. 10. If we consider only the past nearest neighbor [see Fig. 10(b)], then the corresponding entropy θ_{q1} is defined by

$$\theta_{q1} = -\frac{1}{2}\sum_{i=1}^{K}\sum_{j=1}^{K} p(x_i, x_j)\log_2(x_i, x_j). \tag{12}$$

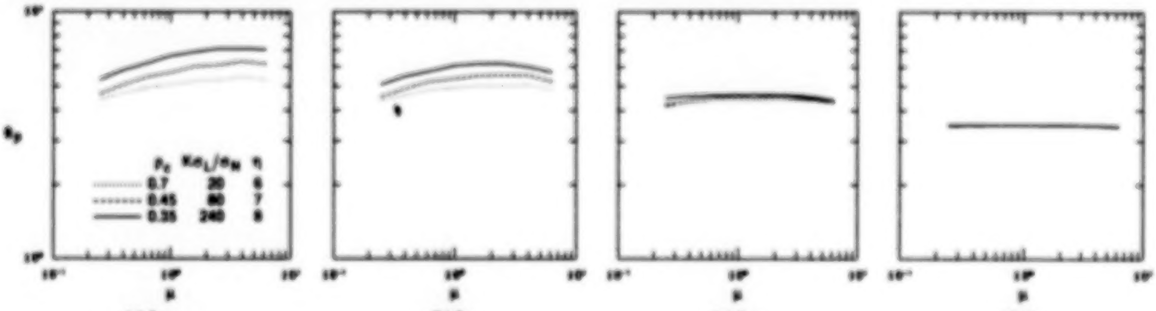


(a) Independent points, θ_{p0} (b) Past nearest neighbor, one dimensional, θ_{p1} (c) Past nearest neighbors, two dimensional, θ_{p3} (d) Nearest neighbors, two dimensional, θ_{p8}

Figure 10. Samples used to estimate the entropy of the digital data $s_q(x, y)$.

The entropies θ_{q3} and θ_{q8} that include, respectively, the past three nearest samples [Fig. 10(c)] and the 8 nearest samples [Fig. 10(d)] are defined similarly. Hence, as we include an in creasing number of neighbors, we approach the entropy θ_q defined by Eq. (10).

Figure 11 illustrates the variation of the entropies θ_{q0}, θ_{q1}, θ_{q2}, and θ_{q3} with the mean spatial detail μ. The zeroth order entropy θ_{q0} does not account for any of the correlation that exists among the neighboring samples. It depends solely on the variance of the incident radiance field and on the effects of the image-gathering process (including quantization). The magnitude of the higher order entropies decreases as more of the correlation among the neighboring samples is accounted for. The independence of the higher entropies from the properties of the radiance field and image-gathering system are probably limited to the informationally optimized designs considered in this paper.



(a) θ_{q0} (b) θ_{q1} (c) θ_{q2} (d) θ_{q3}

Figure 11. Estimates of the entropy θ_q versus the mean spatial detail μ. The image-gathering process is characterized by the optical design parameter ρ_c, the SNR $K\sigma_L/\sigma_N$, and the encoding level η.

Another limitation of these results is that the absolute magnitude of the entropies shown in Fig. 11 cannot be compared strictly to the information density h_q computed by Eq. (9) and shown in Figs. 8 and 9. The reason is that the entropies are obtained from a single target, such as shown in Fig. 3, in which the rectangles have a fixed orientation, whereas the information densities are computed for a circularly symmetric Wiener spectrum derived with the assumption that the orientation of these rectangles is random with a uniform distribution. Hence, the estimates of information efficiency given below will err on the high side. The reason is that the entropy would be higher for the actual radiance field with random edge orientations than for the simulated radiance field with a fixed edge orientation.

B. Image Coding

Figure 12 illustrates the lossless data compression method that we use to assess the effect of compression on the information efficiency of the transmitted data. The purpose of the compression is to translate the string of quantized data $s_q(x,y)$ into an encoded string of data $\Delta s_e(x,y)$ that is (usually) a compressed version of $s_q(x,y)$.

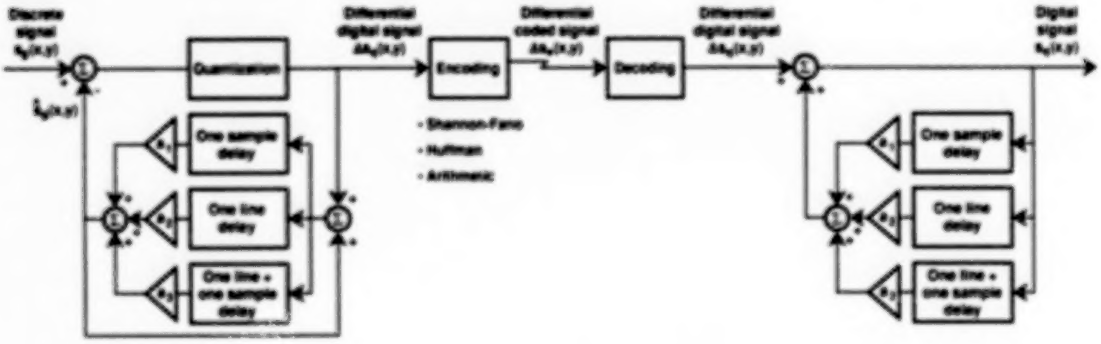

Discrete signal $s_q(x,y)$
$\hat{s}_q(x,y)$
Quantization
Differential digital signal $\Delta s_q(x,y)$
Encoding
Differential coded signal $\Delta s_e(x,y)$
Decoding
Differential digital signal $\Delta s_q(x,y)$
Digital signal $s_q(x,y)$
- Shannon-Fano
- Huffman
- Arithmetic
One sample delay
One line delay
One line + one sample delay
a_1
a_2
a_3
Σ

Figure 12. Model of lossless predictive compression that uses the past three nearest neighboring samples [as depicted in Fig. 10(c)].

The predictive compression reduces the redundancy, or correlation, of the string of digital data $s_q(x,y)$ prior to the encoding. The system shown in Fig. 12 is often referred to as a third-order predictor because it uses the values of the three past nearest neighbors to predict the value that is about to be read out. We select the weighting values a_1, a_2, and a_3 so that the linear mean-square error estimation $E\left\{\left[s_g(x,y) - \hat{s}_g(x,y)\right]^2\right\}$ is minimized. This minimization, which is referred to as best linear estimate, is commonly favored because of its mathematical tractability even though some improvement in performance can often be gained when nonlinear functions are used to form the estimate.[9] For the wide-sense stationary input radiance field, the correlation between neighboring samples is independent of location and the three values a_1, a_2, and a_3 can be computed as follows. Let $R(m,n)$ be the (normalized) correlation between the samples located at $(i-m, j-n)$ and (i,j) [see Fig. 10(c)], then the desired predictor weighting values are given by the following three simultaneous equations:[9]

$$\begin{bmatrix} R(0,1) \\ R(1,1) \\ R(1,0) \end{bmatrix} = \begin{bmatrix} R(0,0) & R(1,0) & R(1,1) \\ R(1,0) & R(0,0) & R(0,1) \\ R(1,1) & R(0,1) & R(0,0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix},$$

where $R(0,0) = 1$.

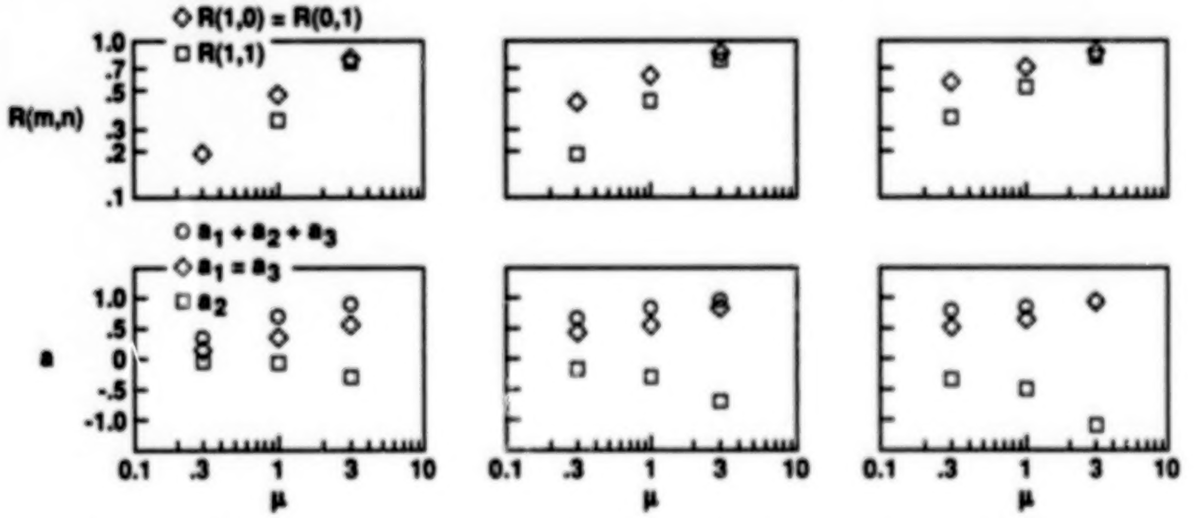
Figure 13 gives the correlation values $R(m,n)$ and the predictor weighting values a_1, a_2, and a_3 for three radiance fields and three image-gathering systems. The radiance fields are characterized by the mean spatial detail μ, and the image-gathering systems are characterized by the optical-design parameter ρ_c, the SNR $K\sigma_L/\sigma_N$, and the number of encoding levels η. The behavior of the correlation and weighting values appeals intuitively. The correlation between neighboring samples increases both as the mean spatial detail becomes larger (i.e., as μ increases) and as it becomes more blurred (i.e., as ρ_c decreases). In the limit, the sum of the weighting values (i.e., $a_1 + a_2 + a_3$) approaches unity, which suggests that the new sample will be similar in value to the neighboring ones with increasing probability. It also is interesting to observe that the correlation $R(0,1)$ and $R(1,0)$ of immediately neighboring samples lies between 0.84 and 0.92 when the mean spatial detail is $\mu = 3$. This result turns out to be in close agreement with the observation made by Gonzalez and Wintz[6] (pg. 298) that, in practice, this correlation typically lies between 0.85 and 0.95 for properly sampled images.

The image coding achieves further compression by transmitting the more probable symbols in fewer bits than the less probable ones. The *Huffman* code[5,19] that we use is derived by successively merging the two least probable samples of $\Delta s_q(x,y)$ into a new sample which is assigned a probability equal to the sum of the former two probabilities. This process is continued until exhaustion. The result of this process is arranged as a tree that is used to determine the code words for the quantized data.

Figure 14 characterizes the effects of the data compression. The compression h_d/h_e is given by the ratio of data density $h_d = \eta$ without coding to data density h_e with coding. This compression does not vary significantly with either the mean spatial detail or the design of the image-gathering system. The compression remains within the range of 1.6 to 1.9, and thus approaches the factor of 2 that is often given for lossless data compression. However, the information efficiency h_q/h_e of the encoded data depends significantly on the image-gathering system design. These results suggest the following generalization:

(4) The upper limit of the information efficiency achieved with lossless data compression increases from ~ 0.5 to 0.9 bif/bit as the information density of the encoded data increases from ~ 2 to 4 bifs. Thus, high information density is transmitted more efficiently than low information density.

This generalization intuitively is appealing since the encoded data contain less image-gathering degradation (e.g., aliasing and noise) when the information density is high rather than low.

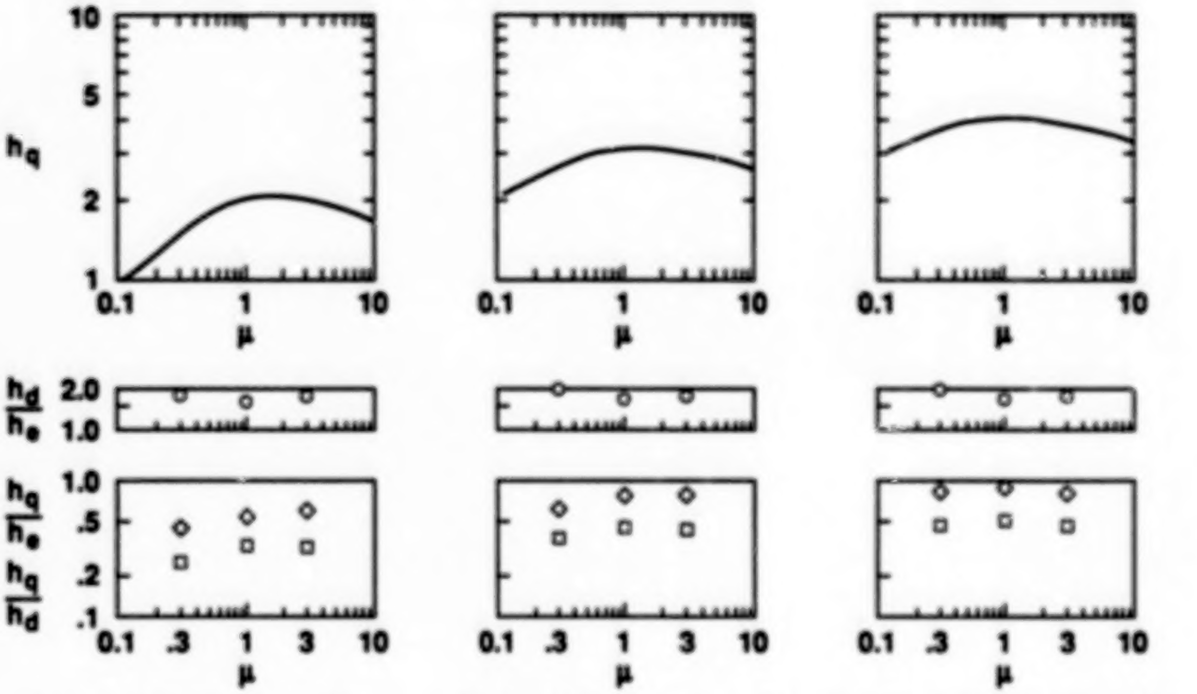


(a) $\rho_c = 0.7$, $K\sigma_L/\sigma_N = 20$, $\eta = 6$

(b) $\rho_c = 0.45$, $K\sigma_L/\sigma_N = 80$, $\eta = 7$

(c) $\rho_c = 0.35$, $K\sigma_L/\sigma_N = 240$, $\eta = 8$

Figure 13. Characteristics of the lossless predictive compressor as a function of the mean spatial detail μ. Shown are the (normalized) correlation $R(m,n)$ between the neighboring samples and the predictor weighting values a.



(a) $\rho_c = 0.7$, $K\sigma_L/\sigma_N = 20$, $\eta = 6$

(b) $\rho_c = 0.45$, $K\sigma_L/\sigma_N = 80$, $\eta = 7$

(c) $\rho_c = 0.35$, $K\sigma_L/\sigma_N = 240$, $\eta = 8$

Figure 14. Characteristic of the encoded signal as a function of the mean spatial detail μ. Shown are the information density h_q, the data compression h_d/h_e, and the information efficiency h_q/h_d and h_q/h_e before and after compression, respectively.

5. IMAGE RESTORATION

A. Information and Fidelity

The data-processing algorithm that maximizes the fidelity of the restored image is given by the unconstrained Wiener filter[1-4]

$$\hat{\Psi}(v,w) = \frac{\hat{\Phi}'_L(v,w)\hat{\tau}_g^*(v,w)}{\hat{\Phi}'_L(v,w)|\hat{\tau}_g(v,w)|^2 \bullet \hat{\text{Ш}}(v,w) + \left(\frac{K\sigma_L}{\sigma_N}\right)^{-2} + \kappa^{-2}}. \tag{13}$$

If the radiance-field spectrum $\hat{\Phi}'_L(v,w)$, the image-gathering response $\hat{\tau}_g(v,w)$, and the SNR $K\sigma_L/\sigma_N$ are exactly accounted for in $\hat{\Psi}(v,w)$, then the image fidelity f reaches its maximum realizable value f_m given by[1-4]

$$f_m = \iint_{-\infty}^{\infty} \hat{\Phi}_L(v,w)\hat{\tau}_g(v,w)\hat{\Psi}(v,w)dv\,dw. \tag{14}$$

It also is possible, then, to express the Wiener filter $\hat{\Psi}(v,w)$ and the fidelity f_m as a function of the spectral information density $\hat{h}_q(v,w)$ as follows:[1,2]

$$\hat{\Psi}(v,w) = \frac{1}{\hat{\tau}_g(v,w)}\left[1 - 2^{-\hat{h}_q(v,w)}\right]$$

and

$$f_m = \iint\limits_{-\infty}^{\infty} \hat{\Phi}'_L(v,w)\left[1 - 2^{-\hat{h}_g(v,w)}\right] dv\, d_w,$$

where $\hat{h}_g(v,w)$ is the integrand of Eq. (9) given by

$$\hat{h}_g(v,w) = \log_2\left[1 + \frac{\hat{\Phi}'_L(v,w)|\hat{\tau}_g(v,w)|^2}{\hat{\Phi}'_L(v,w)|\hat{\tau}_g(v,w)|^2 * \hat{\text{Ш}}_{\neq 0,0}(v,w) + \left(\frac{K\sigma_L}{\sigma_N}\right)^{-2} + \kappa^{-2}}\right].$$

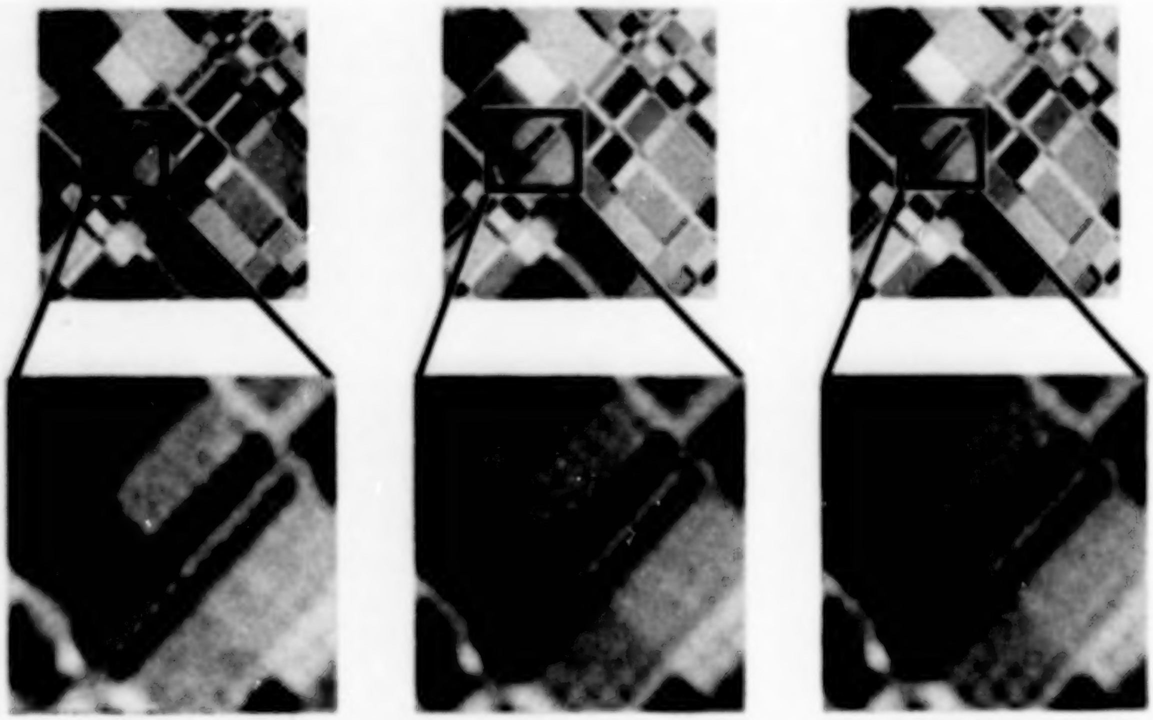
These relationships show that our ability to restore images (restorability) is solely limited by the term $2^{-\hat{h}_g(v,w)}$.

Since the restorability depends on the *spectral* information density $\hat{h}_g(v,w)$ rather than on the *total* information density h_g, it is not possible to directly ascertain whether increases in the information density h_g will always increase the restorability. Nevertheless, it seems reasonable to expect that the restorability of images ordinarily will be correlated positively to the available information density.

B. Fidelity-Maximized Restorations

Figure 15 presents fidelity-maximized images for the three informationally optimized designs characterized in Fig. 14. The change in the visual quality that occurs with increasing information density manifests itself mainly as an increase in the resolution, contrast, and clarity. Noise and aliasing artifacts disappear almost entirely as the highest available information density is approached. However, ringing near sharp edges now becomes a major visual defect. This ringing (Gibbs phenomenon) occurs because of the steep roll-off in the Wiener filter. As Schreiber[5] (pg. 92) summarizes: "... it is impossible to have maximum sharpness with neither aliasing nor ringing, all at the same time. Since the latter is least acceptable, some aliasing and loss of sharpness must be accepted by using a more gradual roll-off in the filter." These results suggest the following important generalization:

(5) Increases in the information density in the fidelity-maximized images is perceived mostly as a decrease in image-gathering degradations (i.e., aliasing artifacts and noise). However, defects (i.e., ringing) introduced by the image-restoration process become more apparent and may limit the amount of information that is useful for image restoration.



(a) $\rho_c = 0.7$, $K\sigma_L/\sigma_N = 20$, $\eta = 6$; $h_q = 2$, $h_e = 3.2$, $h_q/h_e = 0.62$

(b) $\rho_c = 0.45$, $K\sigma_L/\sigma_N = 80$, $\eta = 7$; $h_q = 3.1$, $h_e = 3.9$, $h_q/h_e = 0.79$

(c) $\rho_c = 0.35$, $K\sigma_L/\sigma_N = 240$, $\eta = 8$; $h_q = 3.9$, $h_e = 4.6$, $h_q/h_e = 0.84$

Figure 15. Images restored with the Wiener filter for three informationally optimized image-gathering systems. The systems are characterized by the optical design parameter ρ_c, and SNR $K\sigma_L/\sigma_N$, and the encoding level η. The transmitted data are characterized by information density h_q, data density h_e, and information efficiency h_q/h_e. The mean spatial detail $\mu = 3$.

Figure 16 illustrates the dependence of the image fidelity f_m on the mean spatial detail μ for the three image-gathering systems characterized by Figs. 14 and 15. As can be seen, the image fidelity depends almost solely on the characteristics of the target, even though the resolution, sharpness, and clarity of the images restored for maximum fidelity depend perceptibly on the available information density. This result suggests the following generalization:

(6) Image fidelity is not a suitable criterion for assessing the performance of image gathering and coding. Not only is it insensitive to the visual flaws of the fidelity-maximized images but also to the improvements that are gained in the visual quality of these restorations with increasing information density.

C. Restorations for Visual Quality

The final step in restoring images for maximum visual quality still must be based on perceptual rather than mathematical considerations. For this reason it is necessary to introduce some ad hoc modification of the Wiener filter to control adaptively the trade-off between the enhancement of spatial detail and the suppression of visual defects. Unfortunately, these adaptive controls reduce the quantitative connection between optimum filtering and informationally optimized image gathering that we have tried to maintain so far. This seems unavoidable as long as visual quality cannot be assessed by some figure of merit.

The goals of the ad hoc modification of the Wiener filter are to reduce the ringing at sharp edges and to enhance the visibility of the fine detail and of the boundaries between areas much larger than those that are barely perceived. To provide these adaptive controls, McCormick et al.[4] introduced the Wiener-Gaussian enhancement (WIGE) filter given by

$$\hat{\Psi}_{ie}(v,w) = \hat{\Psi}(v,w)\left\{\exp\left[-2(\pi\sigma_i\rho)^2\right] + \zeta((w\pi\rho)^2\exp\left[-2(\pi\sigma_e\rho)^2\right]\right\}, \tag{15}$$

where ζ is the enhancement parameter that controls the relative amount of the synthetic-high filtered frequency components in the restored image. The standard deviation σ_ζ controls the smoothing of the low-pass filtered image, and the standard deviation σ_e controls the smoothing associated with the edge enhancement. Nevertheless, a trade-off remains between the enhancement of fine detail and sharp edges and the suppression of ringing.

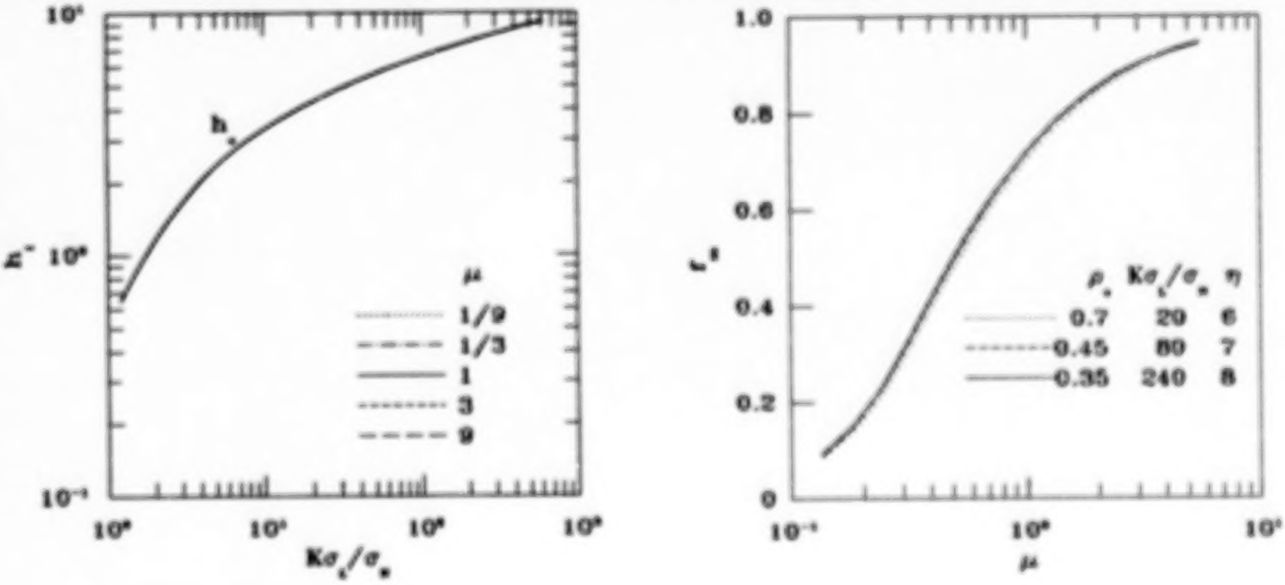
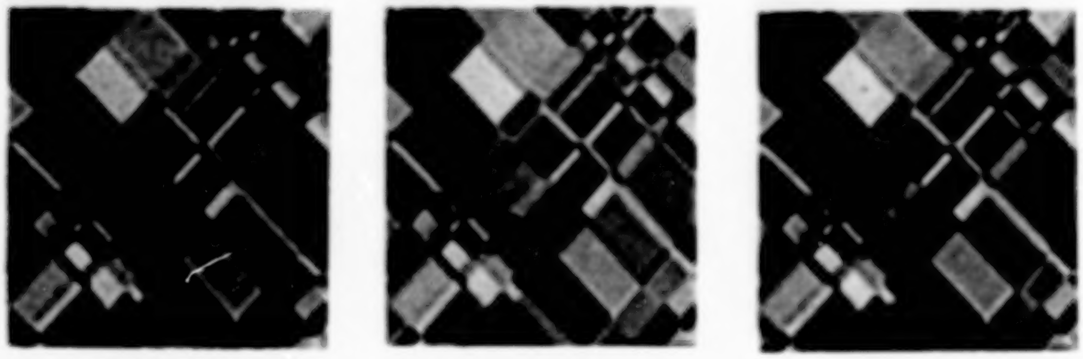


Figure 16. Image fidelity f_m versus the mean spatial detail μ. The image-gathering system is characterized by the optical-design parameter ρ_c, the SNR $K\sigma_L/\sigma_N$, and the encoding level η.

The preferred values for the WIGE parameters depend on the target, the design of the image-gathering system, and the objectives of the observer. For example, if the image-gathering system is informationally optimized for high SNR's and the target is the random radiance field used here as an example, then the trade-off between the suppression of ringing and the loss of sharpness in the fidelity-maximized images shown in Fig. 15 is reasonably well resolved with $\sigma_i = 0.4$ and $\sigma_\zeta = 0.8$. The contrast of the fine detail and sharp edges become enhanced increasingly as ζ is increased. However, this enhancement is achieved only at the cost of general visual quality as well as fidelity. Depending on the objectives of the observer, the preferred value for ζ ranges typically from 0.2 to 0.8.[4]

Figure 17 presents images restored with the WIGE filter. As above, for the fidelity-maximized images shown in Fig. 15, the improvement in image quality with increasing information density is perceived mostly as an increase in clarity. Noise and aliasing artifacts disappear almost entirely as the highest available information density is approached. Moreover, ringing near sharp edges now has been suppressed effectively but at some cost in resolution and sharpness. A small overshoot still occurs at the boundaries between areas much larger than those that are barely perceived. This overshoot enhances the visibility of the boundaries and therefore is often preferred; however, it can be suppressed by reducing the enhancement parameter ζ. These results suggest the following generalization:

(1) The visual quality that can be attained with interactive image restoration improves perceptibly as the information density increases to ~ 3 bifs. However, the perceptual improvements that can be gained with further increases in information density are very subtle.



(a) $\rho_c = 0.7, K\sigma_L/\sigma_N = 20,$
$\eta = 6; h_q = 2, h_e = 3.2,$
$h_q/h_e = 0.62$

(b) $\rho_c = 0.45, K\sigma_L/\sigma_N = 80,$
$\eta = 7; h_q = 3.1, h_e = 3.9,$
$h_q/h_e = 0.79$

(c) $\rho_c = 0.35, K\sigma_L/\sigma_N = 240,$
$\eta = 8; h_q = 3.9, h_e = 4.6,$
$h_q/h_e = 0.84$

Figure 17. Images restored with the WIGE filter for three informationally optimized image-gathering systems. The conditions are the same as in Fig. 15.

The relationship between the available information density and the visual quality of the interactively restored images can be expected to depend significantly on the properties of the target. Hence, this relationship still must be assessed for a variety of different targets and enhancements.

6. CONCLUDING REMARKS

The goal of data compression, as it is traditionally stated, is to reduce, as much as possible, the number of bits necessary to reconstruct a faithful duplicate of the original picture. The effects of data compression are assessed qualitatively by visually comparing the duplicate to the original picture. This assessment ignores entirely the degradations that image gathering and reconstruction introduce into the original picture. It also ignores the potential capabilities of digital processing to reduce the visibility of the degradations caused by image gathering and reconstruction and to enhance various features for close scrutiny.

Our point of view is closer to the one that Schreiber[10] expressed with the question: "For a given channel, what relationship between the original scene and the transmitted signal produces the 'best' picture?" Clearly, this relationship depends on the combined performance of image gathering and coding. However, our constraints, aside from the communication bandwidth, differ from those of Schreiber. Schreiber was concerned mostly with telephotography and televison. These applications are constrained, for commercial reasons, mostly by the cost of the image display. By contrast, we are concerned with space activities and planetary exploration. These applications, in turn, are constrained mostly by the size, weight, and power limitations imposed on the spacecraft instrumentation. The complexity of the digital processing required to restore images and enhance features is ordinarily not a critical constraint. The latter situation also arises frequently in military reconnaissance and medical diagnosis.

Thus, the goal of this paper has been to develop a method for assessing the combined performance of image gathering and coding in terms of the information density and efficiency of the transmitted data. This method is based on earlier findings[1,2] that informationally optimized image gathering maximizes the fidelity of the images restored by the Wiener filter, provided that this filter accounts correctly for the image-gathering degradations. However, an important obstacle remains: the fidelity-maximized images exhibit some visual defects such as ringing, aliasing artifacts, and noise.[3] Therefore, in practice, it is often desirable to reduce these defects with interactive processing.[4]

The method for optimizing the end-to-end performance of image gathering and coding for interactive restoration that is suggested in this paper is (1) to assess how much information is required to restore and enhance images with sufficiently high visual quality for a particular application, and (2) to assess how this information can be acquired and encoded most efficiently. The preliminary results that we have presented are limited to a single, artificial target. However, these results intuitively are attractive and consistent with practical experience.

REFERENCES

1. F. O. Huck, C. L. Fales, N. Halyo, R. W. Samms, and K. Stacy, "Image gathering and processing: Information and fidelity," J. Opt. Soc. Am. A2, pp. 1644-1666 (1985).
2. F. O. Huck, C. L. Fales, J. A. McCormick, and S. K. Park, "Image-gathering system design for information and fidelity," J. Opt. Soc. Am. A5, pp. 285-299, March 1988.
3. C. L. Fales, F. O. Huck, J. A. McCormick, and S. K. Park, "Wiener restoration of sampled image data: End-to-end analysis," J. Opt. Soc. Am. A5, 300-314 (1988).
4. J. A. McCormick, R. Alter-Gartenberg, and F. O. Huck, "Image gathering and restoration: Information and visual quality," J. Opt. Soc. Am. A6, 987-1005 (1989).
5. H. C. Andrews and B. R. Hunt, Digital Image Restoration (Prentice-Hall, Englewood Cliffs, N. J., 1977).
6. R. C. Gonzalez and P. Wintz, Digital Image Processing (Addison-Wesley, Reading, Massachusetts, 1977).
7. W. K. Pratt, Digital Image Processing (Wiley, New York, 1978).
8. T. S. Huang, ed., Picture Processing and Digital Filtering (Springer-Verlag, Berlin, 1979).
9. A. Rosenfeld and A. C. Kak, Digital Picture Processing (Academic, New York, 1982).
10. W. F. Schreiber, Fundamentals of Electronic Imaging Systems (Springer-Verlag, Berlin, 1986).
11. C. E. Shannon, "A mathematical theory of communication," Bell Syst. Tech. J. 27, 379-423, and 28, 623-656 (1948); C. E. Shannon and W. Weaver, The Mathematical Theory of Communication (U. Illinois Press, Urbana, 1964).
12. P. B. Fellgett and E. H. Linfoot, "On the assessment of optical images," Philos. Trans. R. Soc. London 247, 369-407 (1955).
13. J. W. Modestino and R. W. Fries, "Edge detection in noisy images using recursive digital filtering," Comput. Vision Graphics Image Process. 6, 409-433 (1977).
14. Y. Itakura, S. Tsutsumi, and T. Takagi, "Statistical proper ties of the background noise for the atmospheric windows in the intermediate infrared region," Infrared Phys. 14, 17-29 (1974).
15. M. Kass and J. Hughes, "A stochastic image model for AI," IEEE Int. Conf. on Systems, Man, and Cybernetics, 369-372 (1983).
16. C. B. Johnson, "A method for characterizing electro-optical device modulation transfer functions," Photogr. Sci. Eng. 15, 413-415 (1970).
17. A. I. Khinchin, Mathematical Foundations of Information Theory (Dover, New York, 1957).
18. R. C. Jones, "Information capacity of a beam of light," J. Opt. Soc. Am. 52, 493 (1962).
19. D. A. Huffman, "A method for the construction of minimum redundancy codes," Proc. IRE 40, No. 9, 1098-1101 (September 1952).

DATA COMPRESSION FOR THE MICROGRAVITY EXPERIMENTS

Khalid Sayood[†‡], Wayne A. Whyte, Jr.[*],
Karen S. Anderson[†], Mary Jo Shalkhauser[*], and Anne M. Summers[†]

[†]Department of Electrical Engineering
and
The Center for Communications and Information Sciences
University of Nebraska–Lincoln
Lincoln, NE

[*]Space Electronics Division
NASA Lewis Research Center
Cleveland, OH

1.0 BACKGROUND

NASA has undertaken an advanced technology development program in the area of high resolution high-frame-rate video imaging to support microgravity science and applications experiments, with the goal of removing constraints on the amount of high speed, detailed data that can be recorded and transmitted. Numerous microgravity experiments have been proposed for Space Station Freedom and the Shuttle which require a broad range of imaging capabilities. Figure 1 presents survey results of user requirements; the chart shows frame rate versus image resolution requirements for many of the proposed microgravity experiments. NASA will develop a digital video imaging system that will be capable of fulfilling as many of the requirements as is practicable. (Initial survey requirements from several of the experiments far exceed state-of-the-art video imaging capabilities. Reexamination of those requirements is now taking place.)

A representative experiment, sponsored by scientists at NASA Lewis Research Center, is entitled Nucleate Pool Boiling. The experiment involves heating freon locally by means of passing a large current through a thin gold coating on quartz. At some point the freon begins to boil causing vapor bubbles to form, grow and depart from the surface. Information to be derived from the experiment includes bubble shape, bubble growth, collapse, departure, and motion after departure from the surface. To obtain the desired measurement accuracy from the video image data, a minimum resolution of 500X1000 pixels is required at a desired frame rate of 1000 frames per second. These requirements are typical of the resolutions and frame rates needed in many of the proposed microgravity experiments.

The imaging system development will progress in stages starting with a demonstration breadboard system which can be upgraded as technology advances. The system will consist of a high resolution imaging device (camera/solid state sensor), a high speed video interface, and a mass storage device (dynamic RAM/magnetic tape). The Phase 2 imaging device will be a 1024X1024 pixel-addressable solid-state sensor with an 80 Mpixels per second multichannel scan rate, providing monochrome images with 8 bits gray scale resolution. It will be capable of image subframing in order to trade off

---

[†] Supported by the NASA Lewis Research Center under grant NAG 3-806.
[‡] Supported by the NASA Goddard Space Flight Center under grant NAG 5-916.

field of view for frame rate. (For a 1024X1024 pixel image, the 80 Mpixels/sec scan rate of the sensor would allow a maximum frame rate of 76.29 frames per second, however, a subframed image of 128X128 pixels could achieve a frame rate in excess of 4800 frames per second.) The high speed video interface will provide synchronization and routing of data to the mass storage devices. Both high capacity magnetic tape and high speed 512 Mbyte dynamic RAM will be available for mass storage of image data.

Video data compression is scheduled to be incorporated into the imaging system to enhance its capabilities for data acquisition, storage and transmission. Researchers at NASA LeRC and the University of Nebraska–Lincoln are working together to develop appropriate compression algorithms. The data compression aspects of the high resolution high-frame-rate video technology (HHVT) project will be the focus of this paper.

2.0 DATA COMPRESSION IN THE HHVT SYSTEM

The quantity of image data that will be generated by most, if not all, of the proposed microgravity experiments is so large that data compression (image processing) will be a necessity in the imaging system. Image data compression can be used to enhance the capabilities of the HHVT system in at least two areas. First, compression that is achievable between the imaging device and the mass storage unit directly increases the storage capacity. A two-to-one compression factor would double the amount of storable data, thereby doubling the available experiment time. (The high speed 512 Mbyte dynamic RAM can accommodate just 6.4 sec of data at the full scan rate.) The second area of enhancement is with image data transmission to Earth. Here, data compression can be used to reduce the transmission bandwidth and total time required for transmission. (A third area where it may be possible to apply data compression techniques is in the focal plane, however, this area is not currently under study).

The data compression requirements differ depending on where in the system compression is being applied. Compression prior to mass storage must be kept simple for straightforward implementation due to the high data throughput rate. The techniques used must work in real-time and, typically, should be lossless to maintain complete data integrity. (Lossy schemes may be acceptable for some experimental data requirements, however, lossy schemes are generally more complex and hence more difficult to implement for real-time processing. Additionally, the compression techniques to be incorporated at this stage in the system will be hardware based rather than software. It may not be desirable to have multiple algorithms in hardware due to weight constraints, therefore, a single lossless technique which is universally applicable would be preferred.)

Once the data is in the mass storage device, high speed processing becomes less of a requirement on the data compression system. A much broader range of compression techniques becomes available because implementation can be done in software rather than strictly in hardware. Because processing speed is no longer as critical at this stage of the data handling process, different data compression techniques can be applied to particular experiments in order to take advantage of differing end requirements among the various experiments. For example, the fidelity criterion is experiment dependent. Some experiments may require that quantitative data be measured from the video record, as in the measurement of bubble size. Other experiments may only require qualitative observation of experiment progress to enable control of activity. In the latter case, image resolution may not be as critical as near real-time control. The data compression techniques selected will most likely be experiment dependent and as such will be capable of responding to individual experiment requirements.

Feature and data extraction also offer the possibility for significant reductions in data transmission requirements. If sufficient sophistication can be incorporated into the imaging system to extract the required quantitative data prior to downlinking, only the measurement results may need to be transmitted. For example, rather than transmitting the high resolution image of a bubble to determine its size, only the dimensions would need to be transmitted if that information could somehow be extracted from the data. While feature extraction is more commonly associated with image enhancement rather than data compression, many of the same techniques may be applicable to both areas.

The remainder of this paper shall address several of the algorithms which have been studied or are currently under study for application to data compression in the HHVT system.

3.0 DATA COMPRESSION SCHEMES

The different algorithms presented in this section are elements in a possible "toolkit" of schemes which may be available to the user. The compression scheme presented in Section 3.1 is a lossless coding scheme which is very amenable to real-time hardware implementation. This scheme is therefore a candidate for implementation between the imaging device and the mass storage unit. The remaining algorithms are lossy algorithms and could be used (depending on user requirements) after mass storage.

Several of the lossy algorithms were developed with different applications in mind, but can be adapted for use in the HHVT system. A common property of all the lossy systems is their edge preserving capability. This capability is especially important for the types of images generated by the microgravity experiments, as size and location information is usually derived from edges.

It should be noted that the algorithms presented in this paper do not constitute all the algorithms to be investigated for inclusion in the toolkit. This program is in its initial stages and the algorithms presented in this paper are simply some of the algorithms that currently look promising.

3.1 A DIFFERENTIAL LOSSLESS CODING SCHEME

A high resolution image can be viewed as an image which has been "oversampled". This view leads directly to the inference that there is a high degree of correlation between pixels. The oversampling point of view also automatically discards such pathological cases as images of snow on a TV screen, which can play havoc with any data compression scheme. If we assume a Natural Binary Coding (NBC) or Folded Binary Coding (FBC) scheme, we can also assert that a high degree of pixel to pixel correlation will result in a high probability of the most significant bits of the neighboring bits being identical. A similar argument can be used, with some modification, for other binary coding schemes. The noiseless coding scheme presented in this section takes advantage of this fact to provide compression. It has been motivated by an encoding scheme for tree structured vector quantization [1]. The algorithm functions by comparing the current pixel (byte) value with a reference pixel to obtain a prefix and suffix value for each pixel in the image. The prefix and suffix values comprise the noiseless code for the pixel. In the following we describe the details of both the suffix and the prefix.

The prefix value is the number of MSB (upper bits) in a byte that are identical to the reference pixel. For example:

reference pixel (previous byte) = 11010110
current pixel (byte being coded) = 11011010
prefix value = 4 - - - - (1101)

Before being sent the prefix value is Huffman encoded. A given prefix value is assigned a predetermined Huffman code. A Huffman code is a tree code with varying lengths. Values with higher probabilities of occurrence are given shorter binary codes than values with lower probabilities of occurrence. The prefix value can range from zero to eight. The prefix values zero to eight are assigned Huffman codes generated by that image. Currently, a unique set of Huffman codes (for the prefix values) is being generated for every image. Some examples of Huffman codes are shown in Table 1. Further investigation may be given to using standard sets (same set) of Huffman codes for every image. Initial investigation indicates that more than one set of codes would be needed in order to not decrease the compression ratio. A set for high, medium, and low correlation would most likely be used.

The suffix is the bits of the current pixel that are not identical to the reference pixel minus the most significant bit (MSB) of the nonidentical bits. The MSB (of the nonidentical bits) is not sent because it is obviously the opposite of the reference pixel (otherwise it would be the same as the reference and be included in the prefix value).

The actual data sent for each pixel is the Huffman code for the prefix value and suffix, sent as is (bit for bit). In the previous example, if the Huffman code for 4 is 10 the code sent for the current pixel given would be 10010. Due to the Huffman code (variable length code) and the fact that the suffix length is directly dependent on the value of the prefix, the compressed code sent is a variable length code.

The next problem is to actually transfer the new code. Data is transferred in bytes (eight bits). To get data compression, the codes must be compacted into full bytes. If a byte is used for each code there would be no compression. Therefore, bits are placed into bytes and transferred as soon as a byte is filled.

The decoding is done by reading the bytes bit by bit. The bit(s) are matched against the Huffman codes to determine the prefix value. The Huffman code is currently being sent with the encoded image. If no match is found, another bit is added to the prefix bits and the new set is matched against the Huffman codes. Once a match is found, that many upper bits of the reference pixel are set in the current pixel being decoded. Then the next bit (bit # = 7-prefix value) value is flipped, from that of the reference pixel. Then, according to the prefix value, the suffix bits are set. If the prefix value is four, then the suffix must contain three bits. For example, reversing the first example:

code sent = 1 0 0 1 0
first bit compared = 1 (no match)
add bit, compare = 1 0 (matches prefix = 4)
if, reference pixel = 1 1 0 1 0 1 1 0
set current pixel = 1 1 0 1
flip next bit = 1
set the next three (7-4) bits, suffix = 0 1 0
current pixel = 1 1 0 1 1 0 1 0

The next bit read from the code would be the start of the next prefix value.

The very first pixel of every image is always sent as is and is always the first reference pixel. The first line always sets the reference pixel to be the previous pixel, to the left. For the first pixel on each line the reference pixel is always the pixel directly above the current pixel. These reference pixels are always true no matter how the rest of the image is referenced. To determine the reference pixel for the rest of the image, three different algorithms have been investigated. The first algorithm, REFLEFT, sets the reference pixel, except for the first pixel on each line, to be the previous pixel, the pixel to the left. The second algorithm, REFUP, sets the reference pixel, except for the first line, to be the pixel directly above the current pixel. The third algorithm, THRESH, combined the first two algorithms. The third algorithm flips the reference pixel between above and to the left depending on the threshold value. The threshold value is set at the beginning of the program. If a prefix value drops below the threshold value, the reference pixel is switched (from above to left or vice versa). For example, if the reference pixel currently being used is to the left and the threshold value is three and the current prefix value is two, then for the next pixel, the reference pixel used will be above.

In Table 1 the compression obtained, using several images, for the three different algorithms is presented. For the third algorithm, THRESH, the data is presented using threshold values of three, four, and five. The results obtained by using these algorithms were compared against the commercially available compression program PKARC. PKARC compresses files by optimizing between Huffman encoding, a static Lempel-Ziv-Welch coding scheme, and a dynamic Lempel-Ziv-Welch coding scheme. Thus PKARC provides a good benchmark against which to test this algorithm. Note that as the current approach consists of a single algorithm, it is much simpler to implement than PKARC. The results are also shown in Table 1.

As one can see, the new algorithms provide consistently better compression. There is also a direct relationship between the validity of the oversampling assumption and the compression obtained. The compression obtained for the 384X512 images is in general substantially higher than the compression obtained for the 256X256 images. Among the 384X512 images the IBMAD image has the lowest compression because of the presence of granular noise in the image. This is evident from the IBMAD picture. The granular noise because of its "white" nature violates the oversampling assumption. The oversampling assumption is also violated in a more direct manner in Images 13 through 15, and therefore there is a corresponding drop in compression. Obviously this scheme will perform best for the application for which it has been developed, namely, high resolution images.

As mentioned previously, all our tests have been conducted on relatively low resolution images. We expect substantial increases in performance when we code high resolution images. Noting that going from a 256X256 image to a 384X512 image approximately doubles the compression efficiency, we expect the same kind of performance improvement when going from 384X512 to 1024X1024 images.

3.2 ENHANCED DPCM ALGORITHM

An algorithm has been developed which is based on differential pulse code modulation (DPCM) for simplicity of implementation, but incorporates performance enhancements which result in reconstructed images that are subjectively indistinguishable from the original image at an average rate of 1.8 bits per pixel (bpp). A hardware implementation of the algorithm has been developed and is presently undergoing testing. The algorithm was developed for use with standard NTSC (National Television Systems Committee) video images, and will therefore need to be modified for application to the HHVT imaging system. However, the required modifications should not be major, nor should they affect the performance of the algorithm. In addition to the DPCM, the algorithm incorporates a non-adaptive predictor value, non-uniform quantization and multilevel Huffman coding to significantly improve upon the performance achievable using a standard DPCM approach.

A two-dimensional pixel average is used to generate the predicted value, PV, for determining difference values in the DPCM process, as shown in the block diagram in Figure 2. For the NTSC signal, sampling is done at four times the color subcarrier frequency (14.32 MHz). Neighboring pixels having the same subcarrier phasing relationship as the current pixel are used for the prediction. The difference value, DIF, is calculated by subtracting both the predicted value, PV, and a non-adaptive predictor value, NAP, from the current pixel value, PIX, ($DIF = PIX - PV - NAP$). The function of the NAP is to improve the prediction of the current pixel. The non-adaptive predictor estimates the difference value that would be obtained if just the predicted value were subtracted from the current pixel value ($PIX - PV$). The subtraction of the NAP value from $PIX - PV$ causes the resulting difference value (DIF) to be close to zero. The smaller the DIF, the more efficiently the quantized pixel information can be transmitted due to the use of Huffman coding prior to transmission over the channel. (Huffman coding assigns variable length codewords based upon probability of occurrence.) To reconstruct the pixel, the decoder uses a lookup table to add back in the appropriate NAP value based upon knowledge of the quantization level from the previously decoded pixel.

The development of the non-adaptive predictor was predicted on the likelihood that the difference values of adjacent pixels are similar. The prestored NAP values were generated from statistics of numerous television images covering a wide range of picture content. The NAP values represent the average difference values ($PIX - PV$) calculated within the boundaries of the difference value ranges of each quantization level for the sample images. The use of the NAP results in faster convergence at transition points in the image, thereby improving edge detection performance. The rapid convergence also reduces the total data requirements by increasing the percentage of pixels in the middle quantization levels, where the shortest length codewords are assigned by the Huffman coding process.

The quantizer shown in Figure 2 has thirteen (13) levels. Each level has a quantization value associated with a non-uniform range of difference values. The quantizer provides more levels for small magnitude differences which would result from subtle changes in picture content. The human eye is sensitive to small variations in smooth regions of an image and can tolerate larger variations near transition boundaries where large difference values are more likely to occur. The non-adaptive predictor discussed previously, acts to reduce the difference values thus improving image quality by reducing the quantization error. This is because the non-uniform quantizer results in lower quantization error for small magnitude differences than for large magnitude differences.

The final major aspect of the encoding algorithm is the multilevel Huffman coding process. Huffman coding of the quantized data allows shorter codewords to be assigned to quantized pixels having the highest probability of occurrence. A separate set of Huffman codes has been generated for each of the thirteen quantization levels. The matrix of code sets is used to reduce the number of data bits required to transmit a given pixel. The particular Huffman code set used for a given quantized pixel is determined by the quantization level of the previous pixel. As with the NAP, the Huffman code trees were generated by compiling statistical data from numerous images covering a broad range of picture content. Probability of occurrence data was compiled for each of the thirteen quantization levels as a function of the quantization level of the previous pixel. A separate Huffman code set was then generated based on the probability data of "current" pixels falling into each of the thirteen quantization levels of the "previous" pixels. There is a tendency for neighboring pixels to fall into the same or close to the same quantization level. By recognizing and taking advantage of this fact, the use of the multilevel Huffman code sets provides significant reductions in bits per pixel over a single Huffman code tree because they allow a greater percentage of pixels to be represented by short length codewords.

Due to the predictive nature of DPCM-based schemes, bit-errors on the channel can effect the quality of the prediction of future pixels on a line. This has the subjective effect of producing a visible streak across the reconstructed image from the point of the error to the end of the line. To minimize the propagation of such errors, the algorithm employs line and field resynchronization. In addition, the University of Nebraska has developed an error detection/correction scheme which is directly applicable to this algorithm and offers significant error immunity for minimal data overhead.

3.3 EDGE PRESERVING DPCM

Adaptive Differential Pulse Code Modulation (ADPCM) is a very popular compression technique because it is easy to implement, has low processing overhead, and relatively good fidelity. However, ADPCM image compression is far from ideal. The most obvious drawback is poor edge performance. ADPCM cannot track sudden changes in the image statistics, and this causes substantial edge distortion in the reconstructed image. Some changes in the basic approach are required to reduce edge degradation, while retaining simple, high speed image compression.

We have developed a modified ADPCM scheme which uses a very simple algorithm to prevent edge degradation [2]. The structure of the proposed system is based on the embedded DPCM scheme of Goodman and Sundberg [3]. The new system detects edges and sends extra bits containing edge information. We have shown that substantial improvements in both the subjective and objective edge performance can be obtained using this method [2].

Figure 3 shows the general block diagram of a DPCM system. It works much like Delta Modulation. In fact the basic concept is the same; only the information that cannot be predicted at the receiver is sent. P denotes the predictor and Q the quantizer; s is the value of the k input pixel and p is the predicted value. The difference,

$$e = s - p \tag{1}$$

is the prediction error. This value is quantized, and the quantized value eq is sent to the receiver. The quantizer error, q, can be viewed as an additive noise process.

$$e_q = e + q \tag{2}$$

The quantized error, eq, is fed back to the predictor, added to the current predictor value,

$$\hat{s} = e_q + p \tag{3}$$

and used as input to for the next prediction.

$$p = f(\hat{s}, e_q) \tag{4}$$

The predictor function $f(\hat{s}, e_q)$ is discussed in the following section. A corrupted version of e_q arrives at the receiver.

$$\tilde{e}_q = e_q + c \tag{5}$$

where c is the channel noise. This is added to the receiver's predicted value, and if the predictors at the receiver and transmitter are the same, and the channel noise is negligible $\hat{s}$ and $\tilde{s}$ will be the same. Therefore the reconstructed signal, $\tilde{s}$, and the true signal, s, will differ only by the quantizer noise.

$$\tilde{s} = s + q \tag{6}$$

This basic fact has led to many designs that attempt to minimize quantizer noise. Most of them are application specific, and for the most part they are successful, especially when applied to speech signals. However, the results are not as impressive when applied to image data [4] [5]. The best results have been achieved using adaptive quantizers and/or adaptive predictors. Such systems are usually referred to as Adaptive DPCM, or ADPCM [4].

The predictor function, $f(\hat{s}, eq)$, is chosen so as to minimize the variance of eq. There are many well-known adaptive filter algorithms that can be used to adapt the predictor. We have found that the simple Least Mean Square (LMS) gradient search algorithm is an effective algorithm for adapting the predictor. We have previously shown that edge performance is improved if a pole zero or ARMA predictor is used instead of an all pole or AR predictor. Therefore the adaptive predictor used in the ADPCM system is an ARMA predictor. Both the AR and MA coefficients are adapted using an LMS algorithm. Because of the non-stationary nature of image data, optimization of the gain parameter in the LMS algorithm is not possible. The gain should be relatively small to insure stability. The presence of adaptive zeros also makes the system less susceptible to channel noise.

The quantizer is a two-bit Robust Jayant quantizer [6] [7]. It is a uniform quantizer whose stepsize, (k), is adapted based on the previous sample. The stepsize is expanded if the input falls in the outer quantization levels while it is contracted if the input falls in the inner quantization regions. This algorithm is simple to implement and requires very little computational overhead. Since DPCM is most often used in systems where speed is premium, this method is understandably quite popular. It decreases the quantizer noise; however, it doesn't adapt well enough to solve the edge distortion problem. Simulation results in [2] clearly show the poor edge performance of the ADPCM system. A plot of the quantization noise when encoding a simulated edge shows that the magnitude of q is large near the edge and slowly dies away as the system adapts. The error images obtained in this study clearly show that the quantizer distortion is mainly an edge phenomena.

The first step to improving edges is detecting edges. Once this is done, steps can be taken to alleviate the excess noise. Ideally the edge detection would be simultaneously performed at both the transmitter and the receiver thus eliminating the need for transmitting the edge location. Fortunately the Jayant quantizer is well-suited to this task. The Jayant quantizer is designed to track the variance of the quantizer input by changing its stepsize (k). Since edges are regions where the statistics change rapidly, it follows that the stepsize will expand repeatedly when it encounters an edge. This fact is made use of in the following rule to detect edges:

An edge is detected when the stepsize of the Jayant quantizer expands more than P times in succession, $P > 1$. P should be small to reduce the detection delay; a value of two seems to work well. The output of the edge detector is one when edges are present (that is, the Jayant quantizer stepsize expands more than two times in succession) and zero everywhere else. This detector algorithm, with $P = 2$, was added to the ADPCM simulation and tested using a step input. The results showed that both the receiver and the transmitter simultaneously detect the same edges. As such, no extra information is required to synchronize the detectors.

Now that an effective mechanism for detecting edges at both the transmitter and receiver has been obtained, this information can be used to improve the edge performance of the ADPCM system. The structure used in the current approach is the embedded DPCM structure proposed by Goodman and Sundberg [3]. The embedded DPCM scheme employs an additional or "embedded" quantizer to transmit the quantized quantization error of the DPCM structure to develop a strategy for transmission over noisy channels. In the current approach the embedded quantizer is switched on by the edge detector and remains active for as long as the edge detector declares the edge to be active. During this period the embedded quantizer transmits a quantized version of the ADPCM quantizer error q over a "side channel". This is removed from the ADPCM receiver output $\hat{s}$. Thus during the period that the edge detector declares an edge to be active the reproduction error is $(q - q)$ instead of q. This has the effect of reducing the large quantization error at the edges and preventing edge degradation. As the edge is detected simultaneously at both the transmitter and receiver, the receiver knows when to expect transmission over the side channel and the transmitted quantization error values are synchronized with the reconstructed values at the output of the ADPCM receiver. The issue of exactly how to configure the side channel is not addressed in this work. However, the ability to easily achieve synchronization seems to suggest that configuring the side channel should not be a very difficult task.

The proposed system was simulated using the USC GIRL and USC COUPLE image as the source images. A two-bit robust Jayant quantizer and a pole-zero (ARMA) adaptive predictor of the type described before was used. There was considerable improvement in the edge performance. This was reflected in both objective (SNR) and subjective (perceptual) improvement. The overhead due to the side information was less than half a bit per pixel.

While the use of the Jayant quantizer for edge detection is efficient from the point of view of savings on side information, the current definition of an edge is rather ad hoc. Because of this, the savings in side information during the edge detection process may be offset by the extra side information needed for the edge preserving process. In fact an overhead of around 0.5 bits/pixel for a coding scheme with nominal rate 2 bits per pixel seems rather high. We are currently examining this technique from several points of view. The first is to get a more exact definition of an edge in terms of the Jayant quantizer than the one used in the above study. The second is to examine more conventional edge detection systems including the IDS system proposed by Cornsweet [8] and Huck [9]. These methods would be used to find and extract the edges from the image. The edges could then be coded separately, while the image sans edges could be very efficiently coded using a low rate DPCM system. Finally we are examining the possibility of developing multiquantizer ADPCM schemes where the switching between quantizers with different rates would be performed based on the behavior of the Jayant quantizer.

3.4 A MODIFIED RUN-LENGTH CODING SCHEME

The final algorithm presented here is also a variation of the popular DPCM scheme. Again, one of the objectives is to reduce the excessive edge degradation present in standard DPCM systems. Another objective is to have a system that can operate under situations where a common communication channel is being used by a number of users and thus the available channel capacity may vary over the period of a single transmission. Under such situations the system would be able to reduce the rate in return for accepting a certain amount of distortion. However, the edge fidelity which is the primary objective would still be protected.

The system block diagram is essentially similar to the DPCM diagram of Figure 3 with one important modification. The DPCM encoder output forms the input to a modified run-length encoder. Of course, the inverse operation precedes the DPCM decoder. This system is a variation of the system presented in [10]. The various elements of the system are presented below.

The predictor is a one tap "integer" predictor. The output of the predictor is given by

$$p_n = \lfloor a s_{n-1} \rfloor \tag{7}$$

where $\lfloor . \rfloor$ denotes the "floor" function. The floor function is used so as to force the predictor output to be an integer. This was done to allow the system to be used for lossless encoding.

The quantizer is a uniform quantizer with stepsize Δ which effectively contains an infinite number of levels. This means that the only type of quantization noise present is granular noise. There will be no overload noise at the output of the quantizer. If $\Delta = 1$, the quantizer becomes an identity mapping. An infinite number of quantization levels would generally imply an infinite rate; something we definitely want to avoid. This is done by the use of the modified run-length encoder.

The modified run-length encoder puts out n bit, fixed length, codewords corresponding to 2^n output levels of the quantizer. The lowest output level represented is denoted by the symbol LOW while the maximum valued output level represented is denoted HIGH. Note that

$$HIGH = LOW + (2^n - 1)\Delta \tag{8}$$

If the quantizer puts out a value corresponding to the levels between HIGH and LOW, the corresponding n-bit codeword is transmitted by the modified run-length encoder. If the output value X is greater than or equal to HIGH, then the codeword for HIGH is transmitted and X is replaced by X − HIGH. If the new value of X is less than HIGH then the corresponding codeword is transmitted, or else the codeword for HIGH is transmitted and X is again decremented by HIGH. This procedure is repeated until the value of X falls below HIGH. The modified run-length decoder treats HIGH as a "concatenation symbol". Whenever the codeword corresponding to HIGH is received the decoder begins accumulating the values until a codeword corresponding to a value less than HIGH is received. A similar procedure is used when the quantized value is less than equal to LOW.

The effect of this approach is to raise the instantaneous rate whenever the prediction error is high, which usually occurs at edges. However because there is no overload noise there is none of the edge degradation usually associated with DPCM systems. Also by adaptively changing Δ, the output rate of the coder can be made to match the available channel capacity.

The USC GIRL image was encoded using this scheme. Noiseless coding was achieved at the rate of about 6 bits per pixel. At bit rates above 2.5 bits per pixel there was no perceptual difference between the original and reconstructed images. Below two bits per pixel granular distortion was noticeable in the quasi-constant regions. However there was no noticeable edge degradation.

4.0 SUMMARY AND CONCLUSIONS

In this paper we have attempted to present the environment and conditions under which data compression is to be performed for the microgravity experiment. We have also presented some coding techniques that would be useful for coding in this environment. It should be emphasised that we are currently at the beginning of this program and the "toolkit" mentioned is far from complete.

REFERENCES

1 D.L. Neuhoff and N. Moayeri, "Tree Searched Vector Quantization with Interblock Noiseless Coding," Proc. 1988 CISS, Princeton, NJ, pp. 781–783, 1988.

2 S.M. Schekall and K. Sayood, "An Edge Preserving DPCM Scheme for Image Coding," Proc. 31 Midwest Symp. Circ. Syst., St. Louis, MO, 1988.

3 D.J. Goodman and C.E. Sundberg, "Combined Source and Channel Coding for Variable-Bit-Rate Speech Transmission," Bell Syst. Tech. J., Vol. 62, pp. 2017–2036, Sept. 1983.

4 N.S. Jayant and Peter Noll, Digital Coding of Waveforms, Prentice-Hall, New Jersey, 1984.

5 K. Sayood and S. Schekall, "Adaptive Prediction Algorithms in Differential Encoding of Images," Proc. 29 Midwest Symp. on Circuits and Systems, Lincoln, NE, 1987, pp. 415–418.

6 N.S. Jayant, "Adaptive Quantization with a One-Word Memory," Bell System Tech. J., pp. 1119–1144, Sept. 1973.

7 D.J. Goodman and R.M. Wilkinson, "A Robust Adaptive Quantizer," IEEE Trans. on Communications, pp. 1362–1365, Nov. 1975.

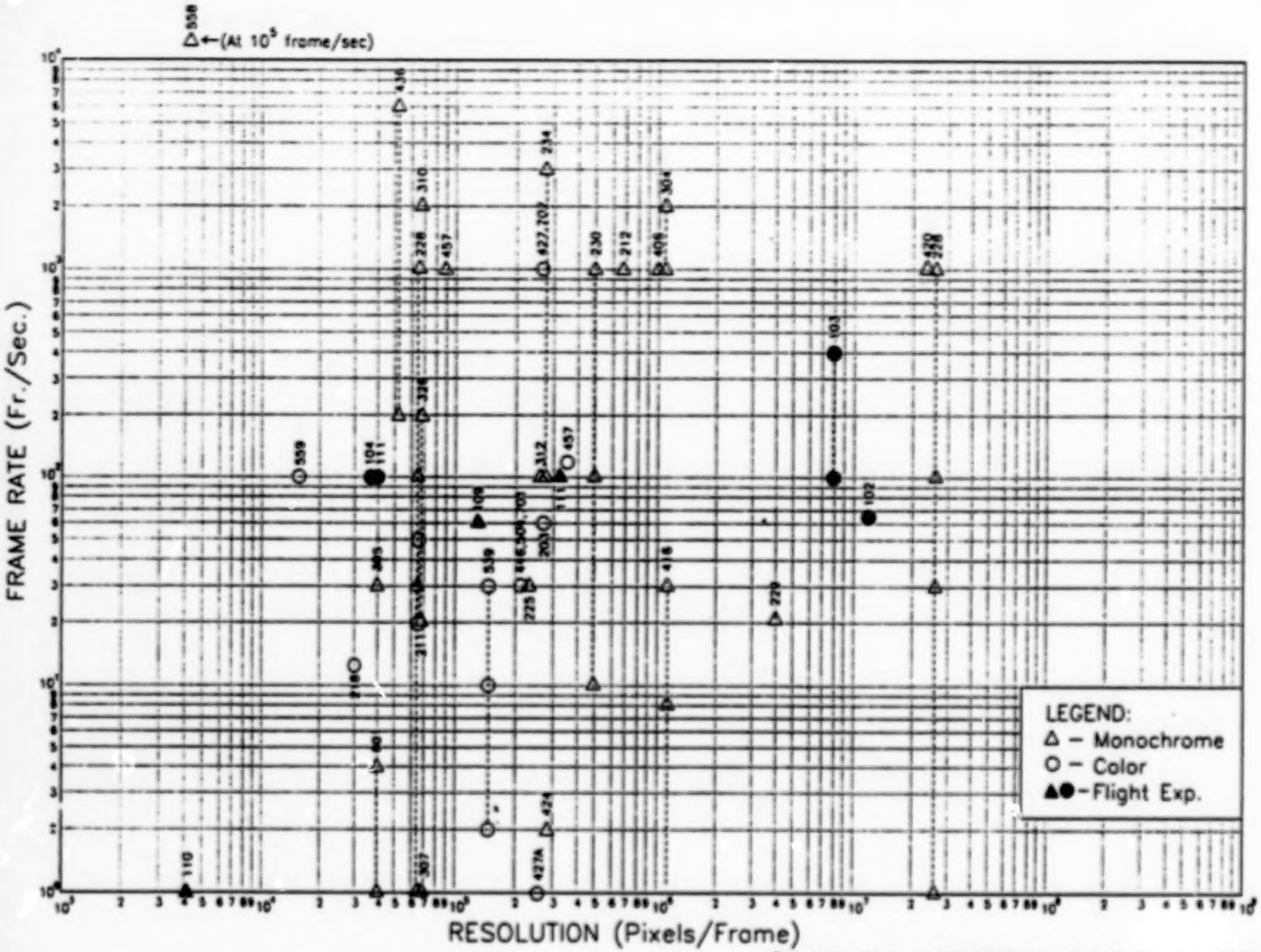
8 T.N. Cornsweet and J.I. Yellot, Jr., "Intensity-dependent Spatial Summation," J. Opt. Soc. Am., pp. 1769–1786, 1975.

9 F.O. Huck, "Local Intensity Adaptive Image Coding," Proc. NASA Science Data Compression Workshop, Snowbird, UT, pp. 301–309, May 1988.

10 K. Sayood and M.C. Rost, "A Robust Compression System for Low Bit Rate Telemetry – Test Results with Lunar Data," Proc. of the Scientific Data Compression Workshop, CP-3025, Snowbird, UT, 1988, pp. 237–250.

TABLE 1

IMAGE	PKARC	REFLEFT	REFUP	THRESHOLD 3	THRESHOLD 4	THRESHOLD 5
(384 x 512)						
IBMAD	13%	23.3%	26.9%	26.7%	26.8%	26.5%
DERIN	33%	45.2%	50.6%	50.8%	50.9%	50.9%
EWEEK	41%	49.3%	53.6%	54.9%	55.1%	55.2%
PATTY	35%	46.1%	46.7%	47.8%	48.1%	48.1%
KARANNE	26%	39.7%	48.5%	47.5%	47.4%	47.2%
MARILYN	30%	41.3%	38.2%	39.6%	40.0%	40.5%
(256 x 256)						
HAT	7%	21.8%	25.0%	23.8%	23.6%	23.4%
IMAGE01	24%	28.3%	28.1%		28.5%	
IMAGE02	27%	33.6%	35.0%		36.3%	
IMAGE03	13%	21.1%	23.7%		22.5%	
IMAGE04	31%	16.0%	16.8%		16.7%	
IMAGE05	7%	15.3%	13.2%		14.8%	
IMAGE06	42%	42.8%	43.1%		43.8%	
IMAGE13	7%	16.3%	14.6%		15.8%	



• Exp. below scale – 110,302,325,413,415,416,427a,501,504,540
See appendix I

Figure 1. User Requirements Survey Results

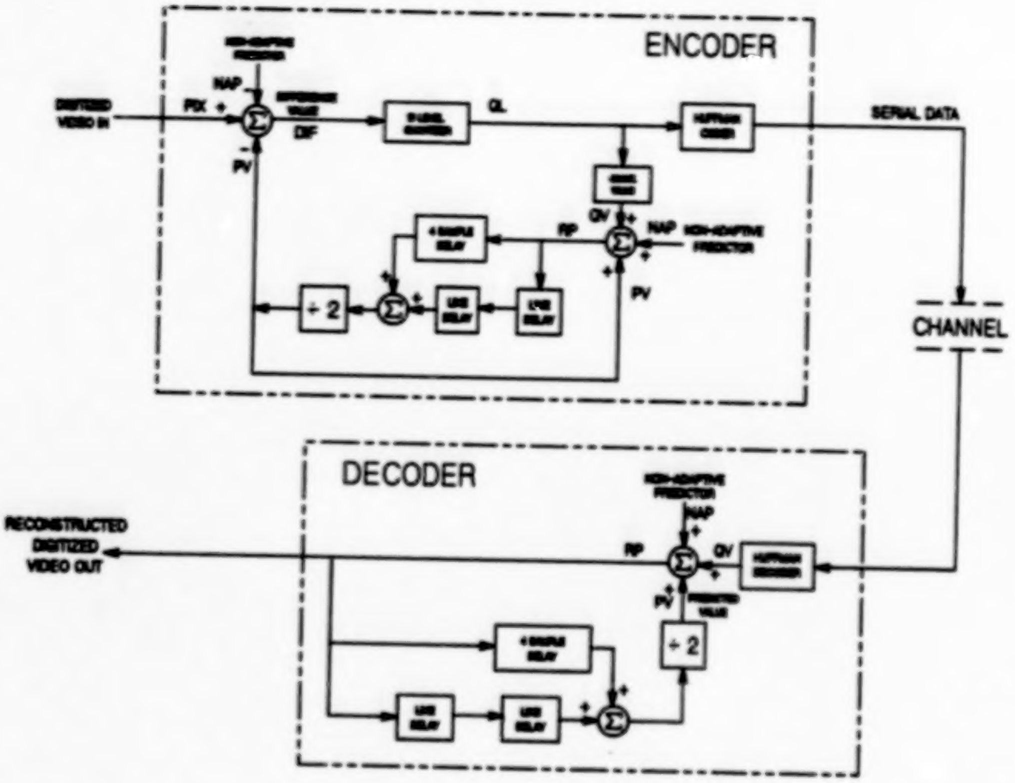
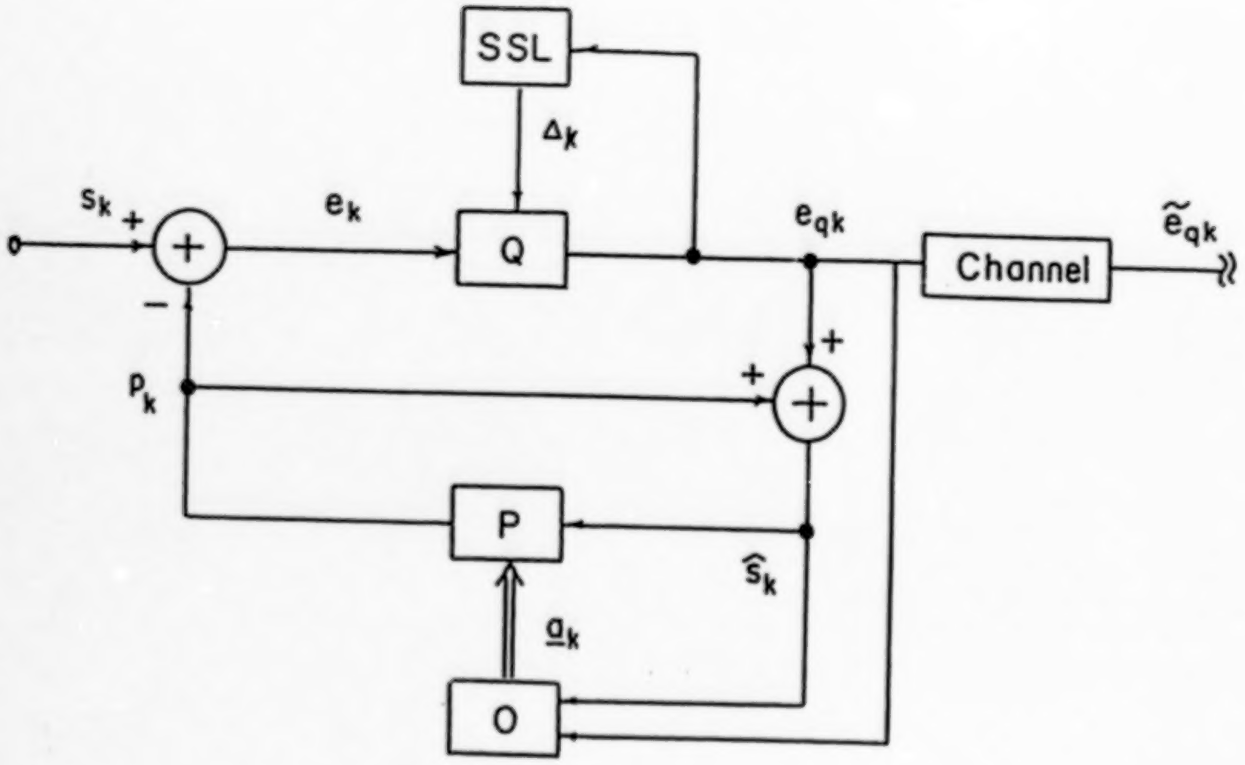
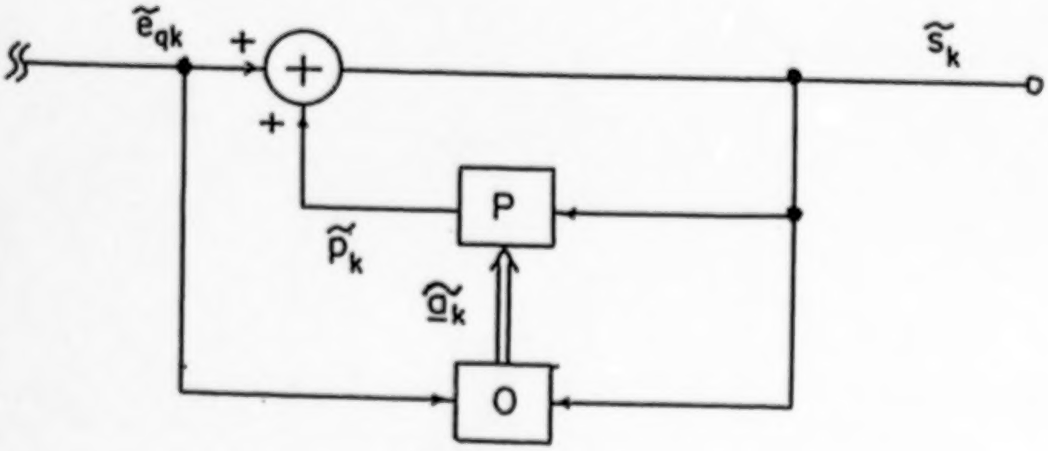

ENCODER
DIGITIZED VIDEO IN
PIX
NAP
DIF
QL
SERIAL DATA
QV
RP
NAP
PV
÷ 2
CHANNEL
DECODER
NAP
RP
QV
PV
RECONSTRUCTED DIGITIZED VIDEO OUT
÷ 2

Figure 2. Enhanced DPCM Algorithm Block Diagram

ADPCM


SSL
Δ_k
s_k
+
−
e_k
Q
e_{qk}
Channel
$\tilde{e}_{qk}$
+
+
p_k
P
$\hat{s}_k$
$\underline{a}_k$
O

Transmitter


$\tilde{e}_{qk}$
+
+
$\tilde{s}_k$
P
$\tilde{p}_k$
$\hat{\tilde{\underline{a}}}_k$
O

Receiver

Figure 3. ADPCM Block Diagram

DETECTION OF EDGES USING LOCAL GEOMETRY

J. A. Gualtieri [1]
Computer Systems Research Facility
Code 635
NASA Goddard Space Flight Center
Greenbelt, MD

M. Manohar [2]
Information Systems Development Facility
Code 636
NASA Goddard Space Flight Center
Greenbelt, MD

1 Introduction

We seek a computational framework for detecting boundaries or edges present in gray level images. We are guided by two notions from psychophysics espoused by Koenderink and van Doorn [1] [2]:

- Primate visual function can be modeled by the activities of locally oriented receptive fields which are the second, third, and possibly fourth order derivatives of the Gaussian of scale t, $\phi_0(t) = \frac{exp(-\frac{(\xi^2+\eta^2)}{4t})}{4\pi t}$.
- In the visual system the natural coordinates on the retina, ξ, η are locally oriented by the direction of the gradient of the image smoothed by $\phi_0(t)$.

The activities of the receptive fields are given by convolving the image with the set of receptive fields at each point in the image, and from this collection of activities at each image point the local geometry is computed. These activities give a convenient representation of the image irradiance and are used to formulate an edge detector.

With the observation that the activities can be related to the Taylor series expansion of the image irradiance, I, about any point in the image we can then give mathematical forms – locally oriented derivatives –for local properties used to model edges and other features.

2 Mathematical Formalism

The receptive fields are denoted ϕ_1, ϕ_{11},ϕ_{12}, ϕ_{22}, ϕ_{111}, ϕ_{112}, …, ϕ_{222} and computed by $\phi_{1\cdots12\cdots2}(\xi,\eta) = \frac{\partial^m}{\partial\xi^m}\frac{\partial^n}{\partial\eta^n}\phi_0(\xi,\eta;t)$, where there are m 1's and n 2's in the subscript of ϕ.

---

[1] National Research Council/Senior Resident Research Associate

[2] National Research Council/Resident Research Associate

While the receptive fields are defined over the infinite plane in actual use we choose a finite support size, $W \times W$, large enough so that the receptive field is very small at the window edge. In Fig. 1 the receptive fields shown are for $t = 2$ with $W = 21$ which give a value of 10^{-4} or smaller for the ratio of the value at the window edge to the maximum value over the entire window.

The receptive fields can be used to compute a finite Taylor series expansion of the smoothed image at each retinal cell, called the jet. The subscripts 1 and 2 denote directions in the local coordinate system along a level contour and along the gradient respectively where the contour direction is given by the convention $\hat{e}_\xi = \hat{e}_\eta \times \hat{e}_I$ where $\hat{e}_\xi, \hat{e}_\eta, \hat{e}_I$ are unit vectors in the direction of increasing ξ, η, I and the cross product follows the right hand rule convention. Fig. 2 shows the coordinate convention. The various receptive fields are named by their subscripts with the convention that the number of 1's is the order of the derivative in the contour direction and the number of 2's the order of the derivative in the gradient direction. The total number of subscripts is the order of the receptive field.

In real images noise and quantitization errors may prevent the Taylor series from being well defined of the image at point (x, y). However, if we use the *derived* image , $I \otimes \phi_0$, given by smoothing the Image with ϕ_0, then we may expand about (x, y) to third order to get

$$\begin{aligned}
&[I \otimes \phi_0](\xi, \eta) = \\
&\quad [I \otimes \phi_0] + \xi \frac{\partial}{\partial \xi}[I \otimes \phi_0] + \eta \frac{\partial}{\partial \eta}[I \otimes \phi_0] \\
&\quad + \frac{\xi^2}{2!} \frac{\partial^2}{\partial \xi^2}[I \otimes \phi_0] + \xi\eta \frac{\partial^2}{\partial \xi \partial \eta}[I \otimes \phi_0] + \frac{\eta^2}{2!} \frac{\partial^2}{\partial \eta^2}[I \otimes \phi_0] \\
&\quad + \frac{\xi^3}{3!} \frac{\partial^3}{\partial \xi^3}[I \otimes \phi_0] + \frac{\xi^2}{2!}\eta \frac{\partial^3}{\partial \xi^2 \partial \eta}[I \otimes \phi_0] \\
&\qquad + \xi \frac{\eta^2}{2!} \frac{\partial^3}{\partial \xi \partial \eta^2}[I \otimes \phi_0] + \frac{\eta^3}{3!} \frac{\partial^3}{\partial \eta^3}[I \otimes \phi_0] \\
&\quad + O(\xi^4, \xi^3\eta, \xi^2\eta^2, \xi\eta^3, \eta^4)
\end{aligned}$$

where all the derivatives and $[I \otimes \phi_0]$ are evaluated at the point (x, y).

For the convolution operator we have

$$\begin{aligned}
\frac{\partial^{m+n}}{\partial \xi^m \partial \eta^n}[I \otimes \phi_0] &= [\frac{\partial^{m+n}}{\partial \xi^m \partial \eta^n} I \otimes \phi_0] \\
&= [I \otimes \frac{\partial^{m+n}}{\partial \xi^m \partial \eta^n} \phi_0] \\
&= [I \otimes \phi\ \mathbf{w}]
\end{aligned}$$

where $\phi\ \mathbf{w}$ is shorthand for $\phi_{1\cdots 12\cdots 2}(\xi, \eta)$. This shows that we may compute activities of the receptive fields and thereby the Taylor series by performing convolutions with the receptive fields. At this point we have made no commitment to the orientation of the local coordinates ξ, η.

3 Choosing the Local Coordinates

Suppose then as it is proposed by Koenderink [2] that the biological visual system makes no commitment as to the coordinate system it will use, but rather chooses to use all coordinate systems. This it does by measuring the receptive fields activities over many directions $-\pi < \theta \leq \pi$. Thus the approximate continuum of quantities (as a function of θ)

$$a^{rot}_{\mathbf{w}}(\theta) = I \otimes R_\theta(\phi\ \mathbf{w})$$

is available to the low level vision system. Here $\mathbf{w}$ is the receptive field name and $R_\theta(\cdot)$ is an operator that rotates its argument through an angle θ. Assuming the image varies smoothly, the maxima and zeroes of these activities define locally meaningful directions. For example, at a particular image point, the angle θ for which $a^{rot}_2(\theta)$ is a maximum defines a local direction with respect to the image x direction that is along the contour, the 1 direction, at that image point. Note for this angle that $a^{rot}_1(\theta) = 0$. Similarly, at a particular image point, the angle θ' for which $a^{rot}_1(\theta')$ is a maximum defines a local direction with respect to the image x direction that is along the gradient, the 2 direction, at that point.

The activities of the receptive fields incorporate the local geometry and provide a useful representation in terms of which to formulate edge detectors. Unlike biological vision, machine vision is typically presented with a much sparser set of activities – those activities of receptive fields defined by derivatives along the image x and y directions. We make contact with biological vision by defining the local coordinate 2 to be in the direction $\theta_{loc} = tan^{-1}(I \otimes \frac{\partial}{\partial y}\phi_0 / I \otimes \frac{\partial}{\partial x}\phi_0)$ and we then compute the activities of the receptive fields in this particular choice of local coordinates:

$$a^{rot}_{\mathbf{w}} = I \otimes R_{\theta_{loc}}(\phi\mathbf{w}).$$

4 Edge Detection and Local Features

A simple edge detector finds candidate edges points as those points where the gradient is a local maximum. The Canny edge detector [3], [4] in doing this pays particular attention to accurately finding the direction of the gradient at each pixel and then to doing a careful interpolation of the change in gradient along this direction so as to find its local maximum.

If the image varies smoothly this is equivalent to locating the zero crossing in the change of the gradient along the gradient direction. In local geometry this is $a^{rot}_1 = 0$, $a^{rot}_2 > 0$, $a^{rot}_{22} = 0$, which means respectively: align 1 along the contour direction, consider only points with nonzero gradient, locate the edge at the zero crossing of a^{rot}_{22}.

For comparison, the Marr-Hildreth edge detector seeks zero crossings of the Laplacian of a Gaussian [5] and is given by $a_{11}^{rot} + a_{22}^{rot} = 0$. This is rotationally invariant indicating that it contains less local geometry than the Canny detector and thereby can be expected not to perform as well as the Canny detector.

Besides exploiting properties in the gradient direction we can compute properties along the contour direction. Any contour of the Image irradiance satisfies implicitly the equation $I(x,y) \otimes \phi_0 = I_0$, where I_0 is value of the smoothed irradiance that defines the contour. Along the contour in image coordinates we have

$$\left.\frac{d}{dx}\right|_c = \frac{\partial}{\partial x} + \left.\frac{dy}{dx}\right|_c \frac{\partial}{\partial y}.$$

Since $I \otimes \phi_0$ is constant on the contour we have $(\frac{d}{dx}|_c)^n I \otimes \phi_0 = 0$, for all n . In particular we have for $n = 1$ and $n = 2$, respectively

$$\begin{aligned} a_1 + \left.\frac{dy}{dx}\right|_c a_2 &= 0 \\ a_{11} + 2a_{12}\left.\frac{dy}{dx}\right|_c + a_{22}(\left.\frac{dy}{dx}\right|_c)^2 + a_2\left.\frac{d^2y}{dx^2}\right|_c &= 0 \end{aligned}$$

If we take the local coordinate system so that the contour lies along the 1 direction, then $\frac{d\eta}{d\xi}|_c = 0$, $a_1^{rot} = 0$, and $\frac{d^2\eta}{d\xi^2}|_c = -a_{11}^{rot}/a_2^{rot}$ which is the curvature along the contour.

Another useful result is obtained by setting $\frac{d\eta}{d\xi}|_c = 0$ in $\frac{d^3}{d\xi^3}|_c I \otimes \phi_0 = 0$ which leads to $a_{111}^{rot}a_2^{rot} - a_{12}^{rot}a_{11}^{rot} = 0$, a result given by Koenderink and van Doorn [1]. Points that satisfy this criterion are called *ridges* and can be identified with corner points along edge directions.

5 Accuracy of the Representation

In accordance with the notion that the local geometry accurately represents the local image structure we expect it should be possible to locate edges to sub-pixel resolution. This approach uses the local geometry to *model* a smoothed representation of the image. Thus the gradient directions are not quantitized by the original pixel lattice (angles quantitized to fall into multiples of $0 < \theta \leq \pi/4$), but are accurately given to a fraction of a degree. Similarly we expect that the location of zero crossings to be given to a finer resolution than an individual pixel.

We have tested this hypothesis by locating a zero crossing along a given direction using an interpolation given by Canny [4]. In Fig. 3 a discretely sampled function h, with $0 < \theta < \pi/4$ is given by values $h(i,j), h(i+1,j)$, and $h(i+1,j+1)$ with $h(i,j) < 0$, and $h(i+1,j), h(i+1,j+1) > 0$. The value $h^{int} = (1 - tan\theta)h(i+1,j) + tan\theta h(i+1,j+1)$

is the linear interpolation of h in the direction θ, and the zero crossing is located at a distance $d = -h(i,j)(1 + tan^2\theta)^{\frac{1}{2}}/(h^{int} - h(i,j))$. Note in this case the zero crossing falls outside the pixel (i,j). Similar interpolation formulas hold for other directions of θ.

In what follows we calculate for each candidate edge pixel the values θ and d. To display the sub-pixel location of the edges we have found, the pixel in which the zero crossing point falls is dilated by a factor of 5 so that each original pixel is equivalent to 25 sub-pixels. The edge is then drawn as a *digital line* so as to pass through the zero crossing at an angle perpendicular to θ. The digital line marks only those sub-pixels that contain the line. We do not extend the line outside the original pixel which contains the zero crossing. This we consider a crude approximation, but the results below bear out the claims of sub-pixel accuracy.

6 Results

We have constructed the local geometry of a simple synthetic image of a rectangle, of step edge with additive Gaussian noise, and of a SAR image (substantially subsampled to remove speckle) taken from SEASAT of ice floes.

Synthetic Image of a Rectangle

For the synthetic image in Fig. 4 the various activities of the receptive fields locate local properties in the image. The zero crossings of the activities of the receptive fields a_{22}^{rot} where $a_2^{rot} > 0$ can be seen to locate the edges of the object while the zero crossings of $a_{111}^{rot}a_2^{rot} - a_{12}^{rot}a_{11}^{rot}$, the *ridge detector* locate the corners. With so little structure in the image it is difficult to give further meaning to the other receptive field activities. The edges found lie to sub-pixel accuracy exactly along the rectangle sides while at the corners they are rounded reflecting the effect of the smoothing by convolution with ϕ_0.

Noisy Step Edges

We created synthetic noisy step edges by adding Gaussian random noise of zero mean and variance one to step edges of varying height. Images were then scaled to the grey scale range of 0 to 255. Defining the signal to noise ratio (SNR) [6] as the ratio of the square step height to the variance of the Gaussian noise we have found, see Fig. 5, that for SNR ≤ 1 the edge becomes broken while for SNR > 1 the edge is continuous. Here the window size and receptive field sizes are, $W = 21$, and $t = 2$. In addition zero crossings of a_{22} are considered only if the values of a_2 exceed $0.1 \max(a_2)$.

As can be seen the edge wanders about the *true* edge but remains smooth and continuous. Roughly then , when the SNR excceeds 2, we expect this edge detector to perform reasonably.

Ice Image

The ice image in Fig. 6 is a 256 × 256 grey scale image dilated in size from a 128 × 128 original image. This was processed using receptive fields of size $W = 21$ and $t = 2$. The dilation was done to reduce numerical errors that would have resulted from using receptive fields of size $W = 11$ and $t = 1$ on the original 128 × 128 image. As for the noisy step edges, zero crossings of a_{22} are considered only if the values of a_2 exceed 0.1 $\max(a_2)$.

Fig. 6 shows I, $I \otimes \phi_0$, a_2^{rot}, a_{22}^{rot} in the top four images and edges located to one pixel accuracy in the lower left. In the lower right are shown the values of d, the distance of the zero crossing from the center of the found edge pixels, coded by intensity with high intensity corresponding to larger d. The large variation in d suggests that the local geometry contains more information that can be used to locate the edge, indeed to sub-pixel accuracy.

To examine the sub-pixel accuracy we have enlarged the upper quadrant of the ice image in Fig 7. As can be seen, the edges found form a smooth almost continuous boundary to the butterfly shaped island and other regions in the image. We take this as evidence that the local geometry contains sufficient information to locate edges to a sub-pixel accuracy that increases resolution by a factor of five.

7 Summary

We have described a new representation, the local geometry, for early visual processing which is motivated by results from biological vision. This representation is richer than is often used in image processing. It extracts more of the local structure available at each pixel in the image by using receptive fields that can be continuously rotated and that go to third order in spatial variation. Early visual processing algorithms such as edge detectors and ridge detectors can be written in terms of various local geomtries and are computationally tractable. For example, Canny's edge detector has been implemented in terms of a local geometry of order two, and a ridge detector in terms of a local geometry of order three.

The edge detector in local geometry was applied to synthetic and real images and it was shown using simple interpolation schemes that sufficient information is available to locate edges with sub-pixel accuracy (to a resolution increase of at least a factor of five). This is reasonable even for noisy images because the local geometry fits a smooth surface –the Taylor series – to the discrete image data.

Only local processing was used in the implementation so it can easily be implemented on parallel mesh machines such as the MPP [7]. We expect that other early visual algorithms, such as region growing, inflection point detection, and segmentation can also be implemented in terms of the local geomtry and will provide sufficiently rich and robust representations for subsequent visual processing.

References

[1] J. J. Koenderink and A. J. van Doorn, *Rperesentation of Local Geometry in the Visual System*, Biological Cybernetics, **55**, 367-375 (1987).

[2] J. J. Koenderink, *Operational Significance of Receptive Field Assemblies*, Biological Cybernetics, **58**, 163-171 (1987).

[3] J. Canny, *A Computational Approach to Edge Detection*, PAMI-8,(1986).

[4] J. Canny, *Finding Edges and Lines in Images*, MIT AI Lab, TR No. 720 (1983).

[5] D. C. Marr and E. Hildreth, *Theory of Edge Detection*, Proc. Roy. Soc. Lond. B **207**, 187-217 (1980).

[6] W. K. Pratt, *Digital Image Processing*, John Wiley and Sons, 495-500 (1978).

[7] J. R. Fisher (ed.), *Frontier of Massively Parallel Scientific Computation*, NASA Conference Publication 2478 (1986).

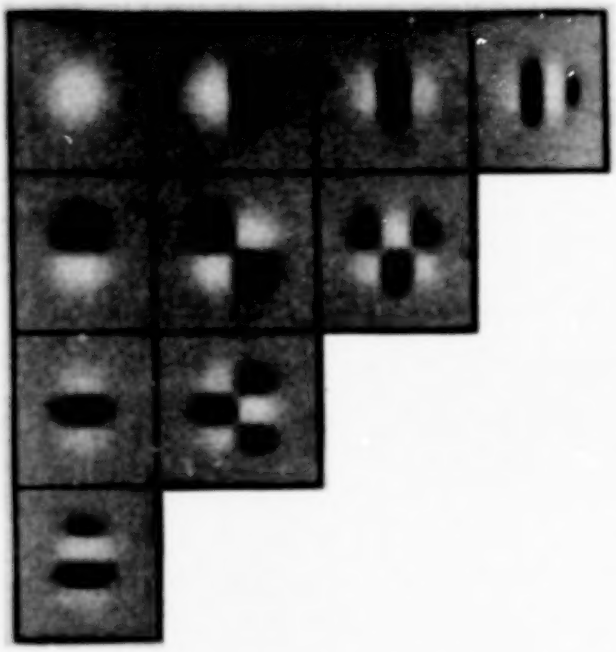


Figure 1: Receptive Fields with $W = 21$ (window size) and $t = 2$ arranged according to:

ϕ_0	ϕ_1	ϕ_{11}	ϕ_{111}
ϕ_2	ϕ_{12}	ϕ_{112}	
ϕ_{22}	ϕ_{122}		
ϕ_{222}			

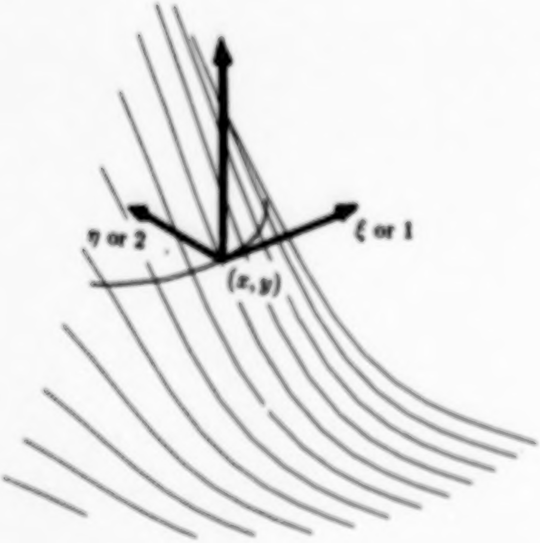


Figure 2: The local coordinates ξ, η (also named 1 2) of the image at a point x, y. The surface shown is the image irradiance versus versus x, y.

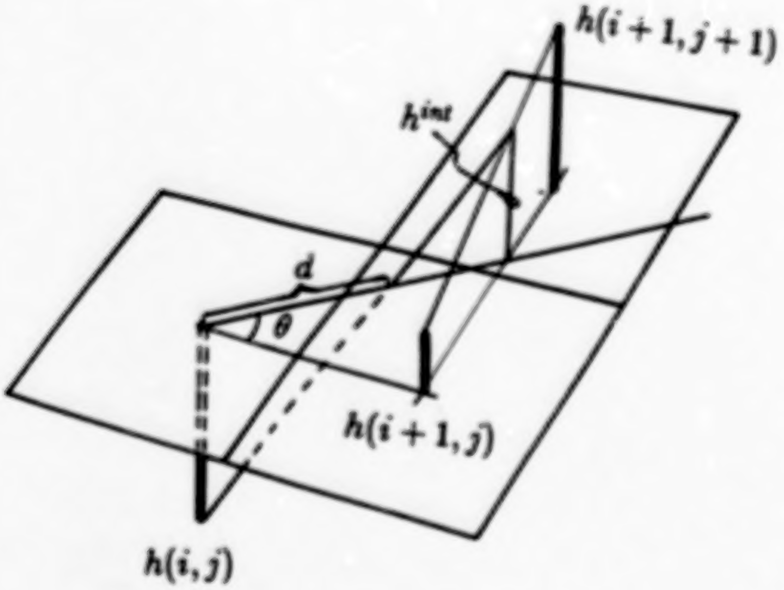


Figure 3: The interpolation scheme used to locate zero crossing.

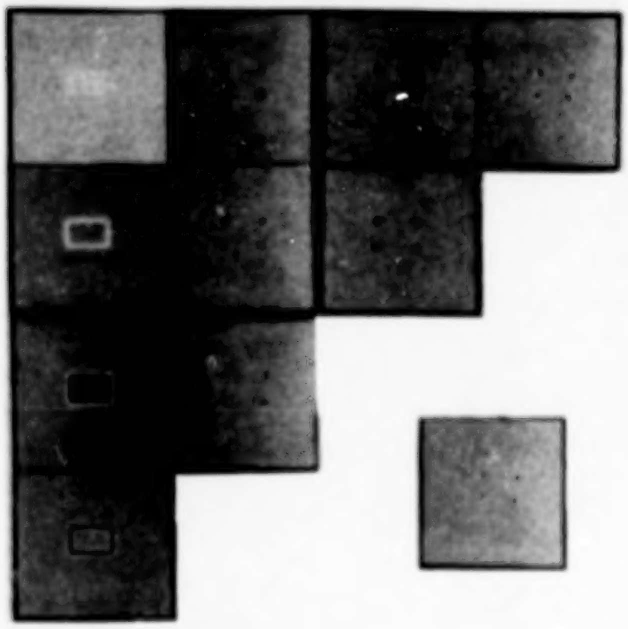


Figure 4: Activities of the receptive fields for the rectangle image, where the size of the rectangle is 64 × 44 pixels, and window size $W = 21$, $t = 2$ according to:

Original Image	a^{rot}_1	a^{rot}_{11}	a^{rot}_{111}
a^{rot}_2	a^{rot}_{12}	a^{rot}_{112}	
a^{rot}_{22}	a^{rot}_{122}		
a^{rot}_{222}			Ridge Points: $a^{rot}_{111}a^{rot}_2 - a^{rot}_{12}a^{rot}_{11} = 0$

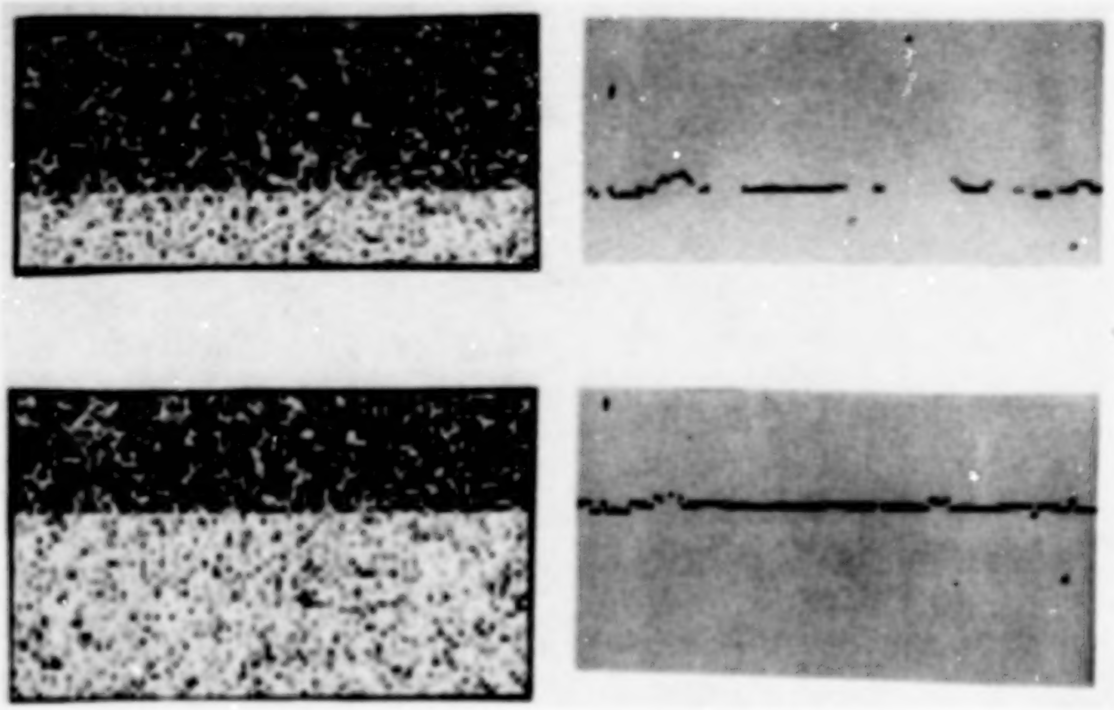


Figure 5: Synthetic noisy step edges on the left and edges found on the right. Upper pair is for SNR = 1 and lower pair for SNR = 2.

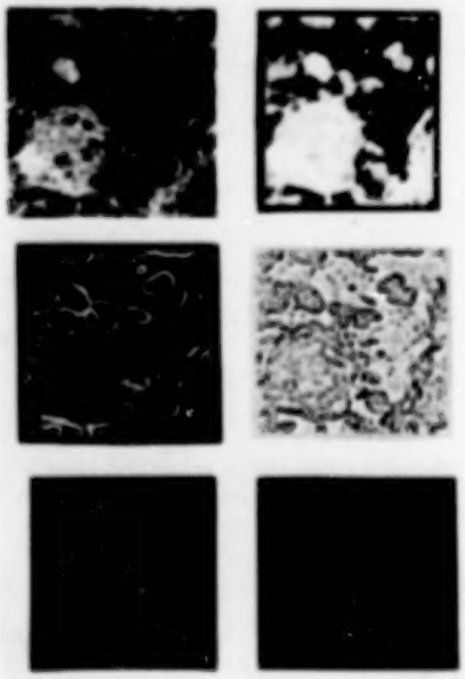


Figure 6: Results for the ice floe image according for $W = 21, t = 2$ according to:

Original Image, I	$I \otimes \phi_0$
a_2^{rel}	a_{22}^{rel}
edge pixels	d for edge pixels

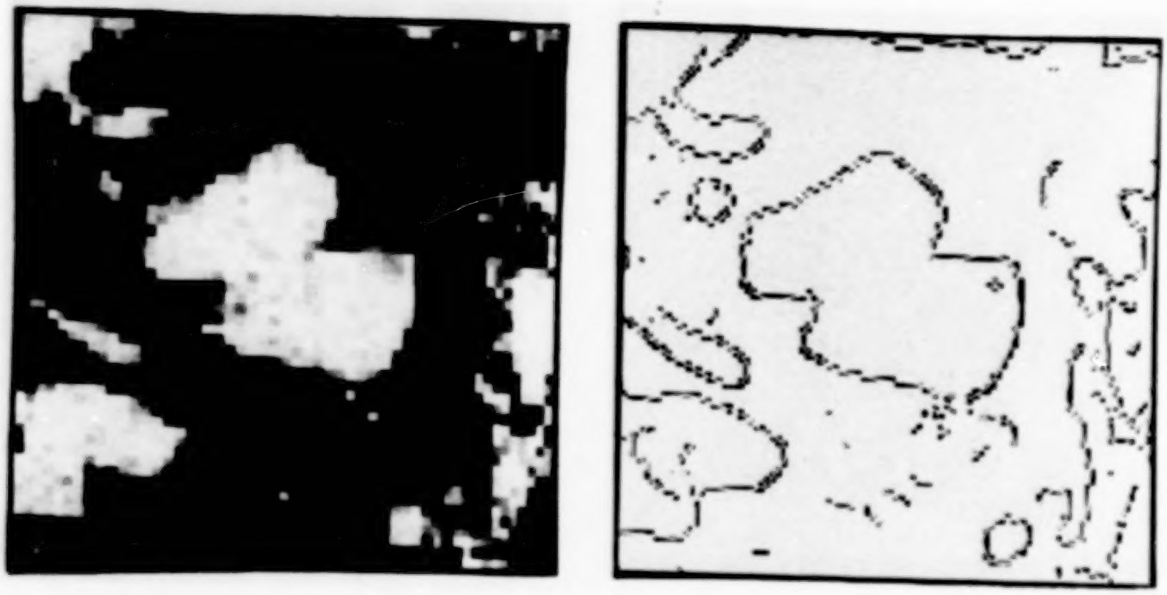


Figure 7: Enlarged view of ice image and edges found.

HIGH COMPRESSION IMAGE AND IMAGE SEQUENCE CODING

Murat KUNT
Signal Processing Laboratory
Swiss Federal Institute of Technology, Lausanne
CH - 1015 Lausanne, Switzerland

ABSTRACT

The digital representation of an image requires a very large number of bits. This number is even larger for an image sequence. The goal of image coding is to reduce this number, as much as possible, and reconstruct a faithful duplicate of the original picture or image sequence. Early efforts in image coding, solely guided by information theory, led to a plethora of methods. The compression ratio reached a plateau around 10:1 a couple of years ago. Recent progress in the study of the brain mechanism of vision and scene analysis has opened new vistas in picture coding. Directional sensitivity of the neurones in the visual pathway combined with the separate processing of contours and textures has led to a new class of coding methods capable of achieving compression ratios as high as 100:1 for images and around 300:1 for image sequences. This paper presents recent progress on some of the main avenues of object-based methods. These second generation techniques make use of contour-texture modeling, new results in neurophysiology and psychophysics and scene analysis.

INTRODUCTION

Every image acquisition system, be it high resolution microdensitometer or TV camera, produces pictorial data by sampling in space and in time, and quantizing in brightness, analog scenes. A digital image is thus an N by N array of integer numbers or picture elements (pixels) requiring N^2 B bits for its representation where B is the number of bits per pixel. This array is commonly referred to as the canonical form of digitized pictures. Generally, the canonical form requires a very large number of bits for its representation. For example, with a 512 by 512 raster and 8 bits per pixel, 2.10^6 bits are needed, a rather large number! For a 3 minute image sequence, at a rate of 25 images per second, the data rate becomes 50.10^6 bits/sec. The goal of image coding is to reduce (to compress), as much as possible, the number of bits necessary to represent and reconstruct a faithful duplicate of the original picture or image sequence. How high a compression can be achieved when a saturation has been reached within the framework of information theory and coding theory ? By simply going out of this framework with the so-called second generation methods [1]. A view of the difference between the techniques of the first and the second generation is the following. Image coding is basically carried out in two steps: first, image data is converted into a sequence of messages and, second, code words are assigned to the messages. Methods of the first generation put the emphasis on the second step, whereas methods of the second generation put it on the first step and use available results for the second step. The very end of almost every image processing system is the human eye. Although our visual system is by far the best image processing system one can think of, it is also far from being perfect. So, if the coding scheme is matched to the human visual system and attempts to imitate its functions, at least for the known part of it, high compressions can be expected. An image can be described in terms of several possible entities such as pixels of the canonical form, a group of pixels in small blocks, Fourier or other transform coefficients, linearly predicted values or derivatives, energy measures within a certain frequency band, etc. With the continuous progress in visual pattern recognition and scene analysis, another possibility is to describe an image in terms of contour and texture [2]. Two main avenues are followed in this paper. The first one imitates some neuronal processing using directional decomposition. The second is based on the segmentation of an image into regions so that region borders fit as much as possible contours of the objects, using either region growing or split and merge.

DIRECTIONAL DECOMPOSITION BASED CODING [3] - [5].

Directional filtering is based on the relationship between the presence of an edge in an image and its contribution to the image spectrum. It is largely motivated by the existence of direction sensitive neurones in the human visual system. A filter whose frequency response covers a sector or a part of a sector in the frequency domain is called a directional filter. To make edge detection with these filters easier, high-pass filtering along the principal direction is introduced. Areas of the Fourier domain corresponding to these filters are shown in Fig. 1. The entire frequency plane is thus covered with n directional filters and one low pass filter. The ideal frequency response of the i-th directional filter is given by

$$H(f,g) = \begin{cases} 1 \ \mathit{if} \ \vartheta(i) \le \tan(\frac{g}{f}) < \vartheta(i+1) \ \mathit{and} \ f^2+g^2 \le \rho_c^2 \\ 0 \text{ otherwise} \end{cases}$$

$$\text{with } \vartheta(i) = \frac{i-1}{2\pi}, \ \vartheta(i+1) = \frac{i+1}{2\pi} \text{ and } |f|, |g| < 0{,}5$$

where f and g are spatial frequencies, ρ_c is the cutoff frequency of the low-pass filter and where a unity sampling step size is assumed. Accordingly, a directional filter is a high-pass filter along its principal direction and a low-pass filter along the orthogonal direction. Because of the Gibbs phenomenon, the ideal frequency response of the directional filters should be modified by an appropriate window function. The purpose is to avoid oscillation around zero crossings corresponding to real edges. One of the most appropriate window functions for this purpose is the Gaussian window. After windowing the filters and filtering, the superposition of all the directional images and the low pass image, lead to the original image. Thus, the directional filtering, as defined, is an information preserving transformation. There are two parameters involved in the directional filters: their number and the cutoff frequency of the low-pass filter. The number of filters is directly related to the minimum width of edge elements that is accepted a priori in the image. Therefore, a direct way to define the number of filters (directions) is obtained by fixing the minimum length of accepted edge elements. From a physiological point of view [6], it seems that the quantization of the directions is made by 20 to 30 different groups of cells, each one specialized to a limited interval of directions. The choice of the cutoff frequency influences only the compression ratio and the quality of the decoded image.

The messages to be coded are the directional images and the low-pass image. Note that the following scheme is not information lossless and that a certain quality degradation is assumed when coding. The main objective is to achieve the highest compression for a given degradation. High frequency images will be used for detecting and coding edges. The loss of information comes from the inevitable choice between weak and strong edges. If the compression ratio is set to high rates, very weak edges must be eliminated. On the other hand, the indirect approximation of the edges by line segments, as assumed by the definition of the edge elements, introduces some degradations at the locations of high curvature. Edge detection in the directional images is based on the high pass character of the directional filters along their principal direction. Filtering a signal with a high pass filter gives zero crossings at the locations of abrupt changes (edges). Accordingly, edge detection in the directional images is performed by searching the zero crossings along the principal direction of each image. The strength of the edges to be retained is controlled by setting a threshold on their slope. Each directional image is represented by the positions and the magnitudes of the zero crossings. The positions are coded with run length coding using the Huffman code, requiring an average of 4.5 bits per position. The magnitudes of zero crossings are coded with three-bit code word.

The low frequency image can be coded in two equivalent ways. Since the maximum frequency of this component is much lower, it can be resampled using the two-dimensional sampling theorem and the resulting pixels can be coded by a standard procedure. The alternative is transform coding. The choice of the transform technique is directly dictated by the filtering that was used. The locations of the Fourier coefficients are known from the characteristics of the filter, and the importance of all these coefficients exclude any elimination by thresholding. This falls, therefore, in the category of zonal coding. After experimenting with several possibilities such as logarithmic quantization, bit allocation plane etc, the coefficients are quantized linearly. Fixed length words are used to code the phase and variable length words, as attributed by the Huffman code, are used to code the magnitudes.

In order to reconstruct the original image, all the components have to be decoded and added. The low frequency component is obtained by inverse transforming the coded coefficients. The high frequency component is obtained by synthesizing the directional images from the zero crossings. The synthesis of edge profiles from the zero crossing information and the interpolation between the columns of the normalized directional images, are the most critical procedures for the quality of the decoded image. An edge model [1] offers the theoretical basis for the synthesis of the one dimensional signals along the edge directions. This model requires two parameters : the magnitude A of zero crossings representing the contrast of the edge and the standard deviation σ related to the steepness of the edge's slope. As the magnitude of zero crossings is coded, the only unknown parameter is the standard deviation. Experimental results indicates that a linear variation of the standard deviation with the contrast gives more realistic edges. The prototype wavelet which was adopted for approximating the profiles of zero crossings is the following:

$$g(u) = \frac{u}{Ak} \exp(-\frac{u^2}{Ak})$$

where u is the distance from the zero crossing at $u = 0$, A the magnitude and k a constant. Once the synthesis of zero crossing profiles is carried out at coded locations, the whole directional image is reconstructed by interpolation between the columns of the subsampled images. For a perfect interpolation between the columns of these images, the fact that the edge elements assumed by the presence of each zero crossing may have any direction within this interval must be taken into account. The interpolation algorithm consists in looking for a neighboring point not only on the same line but also on the two previous or next lines. A first series of decoded images with low compression ratio are shown in Fig. 2. The average compression ratio is around 50 to 1 and the quality of the picture is quite high. By decreasing the cutoff frequency and increasing the zero crossing detection threshold, a second series of results are obtained with higher compression ratios as shown in Fig. 3.

A new step can be made to decrease the redundancy of information by relating the directional images or edges to a prediction model [7]. The goal of this approach is to code the prediction coefficients and errors with less bits than the original information, which implies a good choice of the prediction structure. The study of the computational problems related to the solution of large linear systems for prediction error minimization leads to the conclusion that a synthetic model of the information in the directional image must be built to perform the prediction to avoid tremendous computation. Since in directional images the main information is concentrated in edges, a 2-D linear prediction is chosen. A vector is associated to each edge element, including its position and local profile parameters (magnitude and width) as its components. Then, the prediction model operates on these vectors and estimates edge position and parameters within a prediction structure defined on a set of connected edges.

SEGMENTATION BASED CODING[8]-[11].

In the first stage of this method, the image is segmented to classify its pixels into contour pixels and texture pixels. This procedure partitions the image into a set of adjacent regions under the constraint that the variation of the grey level within the region does not contain any sharp discontinuities, i.e. contours. Segmentation is carried out in three steps: preprocessing, region growing and elimination of artifacts. The preprocessing is intended to reduce the local granularity of the original image without affecting its contours, so that not too many small regions are obtained after region growing. The mechanism of region growing is the following. Regions to be extracted must be characterized with some property in the first step. The property might be, for example, the grey level of a pixel, the variation of the grey level, or the energy within a given frequency band. The selection of this property plays a very important role in the complexity of the method and in the exactness of the contours obtained after segmentation. Then, starting with a given pixel in the picture, its neighbouring pixels are examined to see whether they share the same property. If this is the case, that pixel is included in the region, and in turn, its neighbouring pixels are examined, and so on. When there are no more pixels left, connected to the region and sharing the same property, the procedure stops and restarts at any other pixel which is not included in the first region. The segmentation is complete when all the pixels of the picture are assigned to some region. The property used in our first attempt was very simple: it was a fixed grey level interval. Although it has a constant width, this interval is made adaptive by moving it up and down on the grey level scale in order to intercept the maximum number of pixels. This displacement is constrained, however, so that previously intercepted pixels always remain in the region. Unfortunately, because of the simple property used, the number of these contours is much higher that that of the objects in the original image. Two possibilties are available to overcome this problem : introduction of some distortions by eliminating insignificant regions and their contours, or the use of a more refined property. The first alternative relies on two heuristics: elimination of the small regions and merging weakly contrasted adjacent regions. Statistical analysis indicates that roughly 70 per cent of the regions have less than 15 pixels. To avoid the creation of holes in the image, these regions are included in one of their adjacent regions. To minimize the corresponding distortion, the enclosing region is chosen as the adjacent region whose mean grey level is the closest to that of the small region to be included. By observing areas of constant luminance gradient in the pictures, it can be noticed that they are subdivided into regions even though there is no real contour. This is due to the property used in region growing which divides the image into regions of fixed grey level dynamic range. The second possibility to decrease the number of regions is thus to merge together adjacent regions whose contrast is below a certain level. The contrast between adjacent regions is defined as the mean grey level difference calculated along their common border.

Contours obtained after segmentation are a part of the messages to be coded. A precise description of contours is essential for the human visual system. In this technique contour coding is carried out as follows. Since regions are closed, contour points along the border of two adjacent regions are described twice, once for each region. Prior to coding, these points are removed from one the regions to be described and coded only once. A new and refined code [12] is used requiring about 1.3 bits per contour point.

The missing part of the messages after contour coding is texture coding. It is carried out in two steps. In the first step, the general shape of the grey level in each region is approximated by a two-dimensional polynomial function. The order of the polynomial is determined as a function of the approximation error and of the cost involved in coding polynomial coefficients. A three dimensional view of these approximations is shown in Fig. 3. In this particular case, the best ('cheapest') approximation is obtained with a first order polynomial function. In the second step, the granularity removed with preprocessing is added back in the form of a pseudo-random noise to render the image more natural and less 'painted by numbers'. Fig. 4 shows the final state of the decoded pictures with compressions ranging from 26:1 to 44:1.

Unfortunately, the straightforward generalization of the above described region growing with a more complex property (higher order approximations) is very cumbersome. For this reason a different approach is introduced to achieve the segmentation. It is based on adaptive split-and-merge [13]-[15]. In the first step, the original image is divided iteratively into a set of squares of various sizes. Image data are approximated over each square. The procedure stops when a quality criterion is reached. In the second step, adjacent squares are merged if their joint approximation is satisfactory.

For each region on which the best approximation in the least square sense will be evaluated, two indices of quality are extracted. The first one is a global measure represented by the least mean square error over the region, whereas the second is based on the measure of errors at contour locations within the region of interest. These locations are extracted from the original image using a valid edge operator, whose result is a control image used to control the overall segmentation process. Assuming a power of 2 dimension for the original picture, the split is performed as follows. Starting with the original image, its L^2 approximation is evaluated with a set of 2-D approximating functions. Whenever the values of the quality indices are beyond their respective acceptance threshold, the initial square is split into four squares of identical size. The same procedure is iterated for every subsquare until the quality measure becomes satisfactory. After the segmentation procedure, the 2-D signal is represented by the location of the different segmented regions and the approximation within each region. In this split process, the shapes of the segmented regions are squares of different size. By taking into account geometrical constraints, the structure of the split graph can be reduced to a quadtree representation [16].

The merge process is used to associate different regions obtained by the split operation in order to obtain a more efficient segmentation of the original image. Any segmentation algorithm requires the definition of an appropriate data structure in order to effectively access and relate the different regions. The data structure chosen to merge various squares obtained by the split algorithm is the Region Adjacency Graph (RAG). This is a classical map graph with each node corresponding to a region and links joining the nodes representing adjacent regions. Only contiguous regions are considered as these should be associated first to insure the connexity of the final segmented regions. The basic idea of the merge algorithm corresponds, first, to assign to every link in the graph a value representing the "degree of dissimilarity" that exists between regions (nodes) that this link connects. This degree of dissimilarity constitutes a quality measurement of the approximation. In a second step, the link that exhibits the lowest degree of dissimilarity is removed and the regions (nodes) it connects are merged into one. The procedure is iterated until a termination criterion is verified. At each merging step, the values associated to the links that previously connected the two nodes to the rest of the graph are recomputed. An example of this operation is presented in Fig. 5. These images are segmented into 49 regions with compression ratios ranging from 42: 1 to 68: 1. Two termination criteria were considered to stop this part of the segmentation: 1) the minimum number of regions of the segmented image and 2) the maximum acceptable dissimilarity between the original image and the approximated one.

As post processing, a smoothing technique can be used to enhance the quality of the segmented image in case of polynomial approximation. This procedure can be used after the split-and-merge. Due to the structure of polynomial functions, the approximated signal between adjacent regions may be discontinuous. This may create "false contours" at the boundaries of some adjacent regions. Between two consecutive crossing points of the region frontiers, just one bit is necessary to represent whether this portion of border corresponds to a false contour or not. Once, it has been established which contours do not correspond to real edges in the original picture, a smoothing algorithm is applied to both sides of this "false contour". The width for which the approximated signal is smoothed with respect to each region is linearly dependent on the number of points of the considered region.

IMAGE SEQUENCE CODING BY SPLIT AND MERGE

Ideally, in object-based image sequence coding one should first determine all the objects in the first frame of the sequence. Then, events like motion, enlargement, modification and the disappearance or appearance of new objects need to be detected and analyzed. There are several techniques that could be employed to try to accomplish these tasks. Segmentation has been successfully used in static object-based coding. It can be obtained in many ways: contour extraction, split and merge, region growing, etc. All these methods do not, of course, give the same results, but they all attempt to extract regions whose frontiers match the borders of the objects in the scene. On the basis of the high performance obtained in the static case, split and merge is applied to image sequences in our current work. Past experience suggests that the coding method be matched to the nature of the data, i.e. to its 3-D character. Therefore, a 3-D split-and-merge algorithm is investigated for low bit rate image sequence coding.

The data we are aiming at compressing result from a digital image sequence, where each image is 256x256 pixels in size, and the images are transmitted at 25 images/sec. There is thus a tridimensional data space (x,y space + time). In the method to be described below, the sequence is divided or segmented into various regions. A region is a set of contiguous volume elements, or voxels, which share one or several properties. In our current work, we define a region as a regular domain in the image sequence, over which the grey level variation can be approximated within a specified error, by a 3-D polynomial. A region can thus be represented by the coefficients of its approximating polynomial and by the description of its 3-D border. To find these regions, we use a 3-D split and merge algorithm, which consists of starting from an initial region space and then merging these regions according to the properties they share.

To get the initial region space, from which the final regions will be grown, the 3-D data space can be split into regions such that the luminance in each region is closely approximated by a 3-D polynomial. The initial regions are obtained in the following way: First, we consider the entire image sequence, trying to approximate it by a single polynomial. As the image sequence does not usually have a homogeneous luminance, its approximation by one polynomial is not usually acceptable, because the resulting approximation error is very high. In that case, the entire data space is split in the three directions (x,y and time) to get 8 smaller volumes. Then, the approximation is performed on each one of those volumes. If the approximation error is too high for a given volume, the volume is split again, and so on. The process stops when the approximating error on each sub-volume is lower than a predefined value. Finally, at the stop level, each sub-volume will be considered as an initial region.

Once the initial region space has been obtained, the most similar regions must be merged together. For that, a region adjacency graph is used as before. As shown in Fig.6, a region adjacency graph (RAG) is a graph in which each node represents a region and each branch represents an interconnecting link between two neighboring regions. A cost is assigned to each branch of the RAG to indicate the similarity of neighboring regions. The more similar the regions are, the lower the cost is. Merging the regions sharing similar properties is performed with a priority order. The regions whose interconnecting cost is minimum are merged first. In this way, we assure that the growth will be homogeneous and isotrópic. By iteratively repeating the merge of the most similar regions, we reduce the redundant information contained in the various region attributes. The merge process stops when no more regions should be merged, either because the interconnecting cost is too high, indicating high dissimilarity of the regions, or because a predefined minimum of regions has been reached.

In our application, the grey level variation over a region is approximated by a polynomial function. The interconnecting error of neighboring regions R_i and R_j is the error that results from the approximation of the image over the joint region R_i » R_j. We have chosen to calculate the approximating error in the least mean squared sense. As regions can have any size, any shape and any grey level variation, the approximating error can, in general, take a large range of different values, from less than 10^{-3} to more than 10^8. To have a smaller range of the possible values for the interconnecting cost, the cost will be defined as a logarithmic function of the approximating error.

The function which has been chosen to approximate the image over each 3-D domain is a polynomial of first degree in x and y and of degree 0 in t. This choice can be justified because the grey level variation in the time direction is often not very significant between two successive frames. The first degree in x and y allows the representation of linear grey level variations over the object. Thus, the interpolation polynomial Q is given by $Q(x,y,t) = c + ax + by$, where a, b and c are the coefficients which must be determined in order to have the best approximation of the image for a given region.

To code the borders of the regions, a pyramidal structure is used, composed of a set of parallelepipeds of various sizes. The domain of each region can be thus represented by a set of parallelepipeds of appropriate dimensions (fig. 7). To represent the border of a region as well as possible, the set of the parallelepipeds must match exactly the region that it defines. By setting some constraints on the dimensions and on the positions of the parallelepipeds in the pyramidal structure, it is possible to get a compact code to describe the borders of all regions.

To obtain the most compact pyramidal structure representation, the pyramid is obtained by splitting the data volume along three different directions: x=constant, y=constant and time t=constant. As each cut performed on a volume give rise to two new sub-volumes, the set of the cuts which are performed on the sequence can be represented using a binary tree, where each branch in the binary tree indicates a subdivision in the x,y or time direction.

To get the shapes of the regions, we have to determine which parallelepiped belongs to which region. For that, a label is assigned to each region, such that two neighboring regions do not have the same label. The parallelepipeds will be labeled according to the region to which they belong. Thus, two neighboring parallelepipeds will have the same label if, and only if, they belong to the same region. It has been shown that 8 different labels are enough to properly label a tridimensional graph. However, although it is theoretically possible to get this minimum of 8 colors, the computer time that would be spent to obtain a maximum of 8 colors could be extremely large. The labeling method we have used does not attempt to reduce the number of the colors to its absolute minimum, but it has the advantage of being very fast. Despite the fact that we do not seek an absolute minimum, the number of colors used is still quite small and has never exceeded 13. Thus, 3 to 3.5 bits in average will be enough to code the label of a parallelepiped.

The coefficients of the approximating polynomials are coded in a straightforward manner, using 16 bits: 5 bits are used for coefficient a and b, and 6 bits are used for the c coefficient. The c coefficient, representing the average grey level, is judged more important than the coefficients a and b, representing the grey level variation along the x and y coordinates. For this reason, the c coefficient has been coded with more precision than the other coefficients.

Tests of the proposed algorithm have been performed using standard image sequences. Image sequences 256x256 pixels in size, transmitted at 25 images/sec have been compressed by a factor of approximately 200 to 300, thus allowing transmission through a 48 to 64 kbits/sec channel. The quality of the restored images is reasonably good. However, additional work is required to remove some side effects of the split and merge and reconstruction processes. The main quality defects are the artifacts located at the border of two regions which have been formed by the merge of large initial regions (large cubes). Another defect in the image quality is the imprecision of the borders of the objects in a scene. This is principally due to the size of the smallest possible initial region, which has been fixed at 2x2x2 voxels, to avoid indetermination, while performing approximating error calculation, in the least-mean-squared sense.

CONCLUSIONS

In this paper a brief overview is given of image coding techniques using the contour-texture model. The compression ratio achievable with these techniques may reach very high values. In contrast with conventional methods using a signal processing approach in the selection of the messages to be coded, they are based on scene analysis features. These new methods put heavy emphasis on the selection of the messages to be coded. In this context, signal processing approaches are not as successful as pattern recognition or artificial intelligence approaches. The coding is done in the classical way. It is clear that the methods we have presented need several improvements to produce better quality images at the same compression ratio or to reach higher compression ratios for the same quality. More detailed results can be found in [17].

Because of computer memory limitations, the split and merge algorithm has not been tested on sequences longer than 32 frames. By performing the segmentation algorithm on a longer sequence (a few seconds in length), the compression ratio could be considerably increased, because of the redundant information contained in successive frames. This remark is valid only if the movements in the scene are not too large, and if the border on the time axis is constrained to the same scene. The compression ratio could also be increased by improving the coding of the polynomial coefficients, by using vector quantization coding rather than fixed length coding.

The goal to be reached is to segment the image into regions corresponding to the real objects of the scene, without missing small ones and without introducing false objects and hence false contours. Powerful representations should be designed to describe the grey level evolution within each region. Recent efforts in texture analysis and synthesis will be of great value to image coding to render the natural look when added to the representation of regions. It is hoped that image coding will remain a center of interest for researchers and that even higher compressions will be obtained.

ACKNOWLEDGMENTS

The author expresses his thanks to his colleague Dr. T. Reed and to his Ph.D. student Mr. P. Willemin for their help in the preparation of the manuscript.

REFERENCES

[1] M. Kunt, A. Ikonomopoulos and M. Kocher, "Second Generation Image Coding Techniques" (Invited Paper), Proc. IEEE, Vol. 73, No. 4, April 1985, pp. 549-574.

[2] M. Kunt, "Edge Detection : A Tutorial Review" Proc. ICASSP 82, Paris, May 2-5, 1982, pp 1172-1176.

[3] A. Ikonomopoulos, M. Kocher, and M. Kunt, "Image Coding Based on Human Visual System Propertyies for Optimal Reduction of Redundancy", Proc. 3rd Scandinavian Conference on Image Analysis, Copenhagen, Danemark, July 12-14, 1983, pp. 216-222.

[4] M. Kunt, A. Ikonomopoulos and M. Kocher (invited lecture) "Compression d'images : Methodes de la deuxie`me generation" Premier Colloque GRETSI-CESTA, Biarritz, France, May 21-25, 1983.

[5] A. Ikonomopoulos and M. Kunt, "High Compression Image Coding Via Directional Filtering" Signal Processing, Vol. 8, No. 2, April 1985, pp. 179-203.

[6] D.H. Hubel and T.N. Wiesel, "Brain Mechanism of Vision", Sci. Amer., Vol. 241, pp.150-162, Sept. 1979.

[7] M. Benard and M. Kunt, "Linear Prediction in Directional Images" Proc. EUSIPCO-86, 2-5 Sept. 1986,The Hague, The Netherlands, North Holland.
[8] M. Kocher and M. Kunt," A Contour-texture Approach to Picture Coding", Proc. ICASSP-82, Paris, May 1982, pp. 436-440.
[9] M. Kocher,"Codage d'images a` haute Compression base sur un modèle Contour-texture" Ph.D. Thesis, No. 476, Dept. of Electrical Engineering, Swiss Federal Institute of Technology, Lausanne, Switzerland, March 1983.
[10] M. Kocher and M. Kunt, "Image Data Compression by Contour-texture Modelling", SPIE Int. Conference on the Applications of Digital Image Processing, Geneva, April 1983, pp. 131-139.
[11] M. Kocher and M. Kunt, "A Contour-Texture Approach to Picture Coding" Proc. Melecon-83, Athens, Gre`ce, May 24-26, 1983, Paper C2.03.
[12] M. Eden and M. Kocher, "On the Performance of a Contour Coding Algorithm in the Context of Image Coding. Part II : Coding and Contour Graphs" Signal Processing, Vol. 8, May 1985, to be published.
[13] R. Leonardi, "Segmentation adaptative pour le codage d'images", Ph.D. Dissertation, No. 691, Dept. of Electrical Engineering, EPFL, July 1987.
[14] M. Kocher and R. Leonardi, "Adaptive Region Growing Technique Using Polynomial Functions for Image Approximation", Signal Processing, Vol. 11, No. 1, July 1986.
[15] R. Leonardi and M. Kunt, "Adaptive split and merge for image analysis and coding" SPIE Int. Symp. Image Coding, Cannes, France, Dec. 2-6, 1985.
[16] Y. Cohen, M.S.Landy and M. Pavel, "Hierarchical Coding of Binary Images", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-7, No. 3, May 1985, pp. 284-298.
[17] M. Kunt, M. Benard and R. Leonardi, "Recent Results in High Compression Image Coding" (invited paper), IEEE Trans. on Circuits and Systems, Vol. CAS-34, No. 11, November 1987.

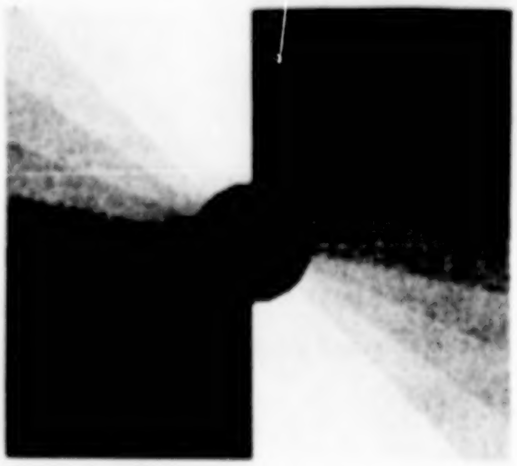
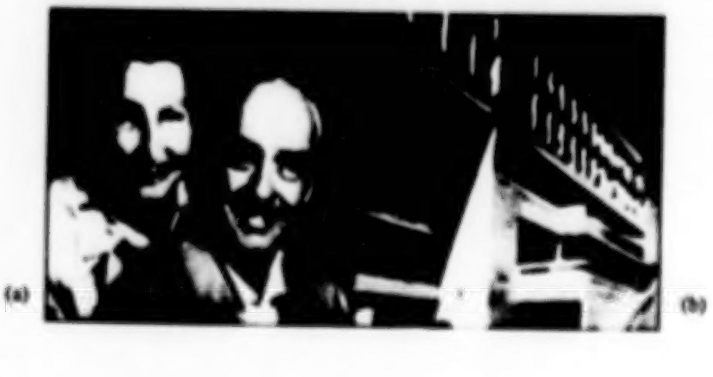


Fig. 1. Sectors of the Fourier domain covered by directional filters.



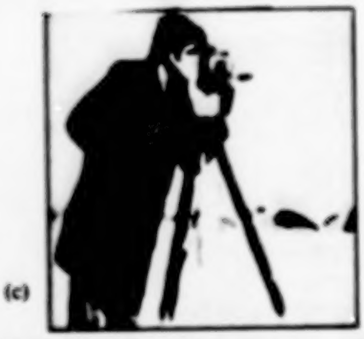


Fig. 2. Directional decomposition based coding results. The compression ratios are 57:1, 59:1 and 49:1 respectively.

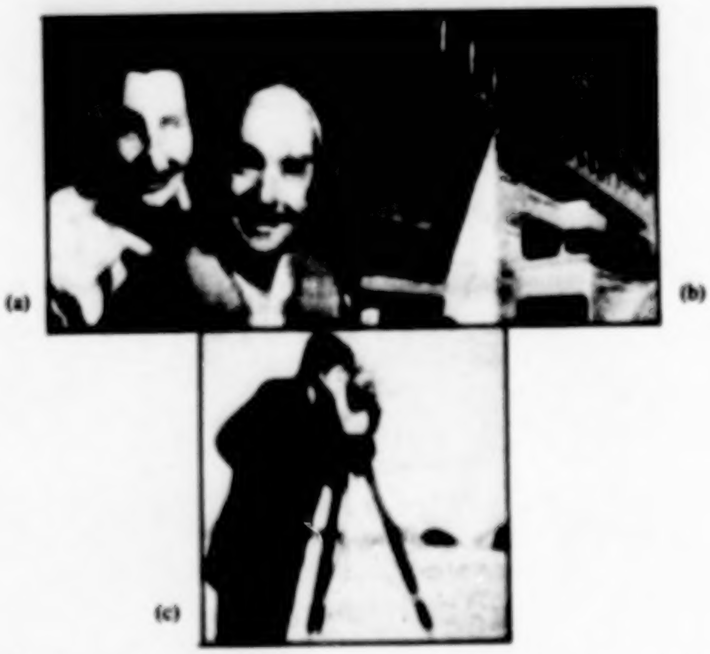


Fig. 3. Directional decomposition based coding results. The compression ratios are 118:1, 86:1 and 84:1 respectively.

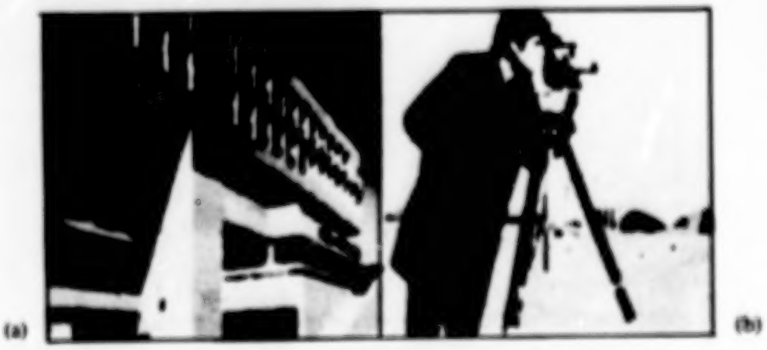


Fig. 4. Region growing based coding results. Compression ratios are 44:1, 38:1 respectively.

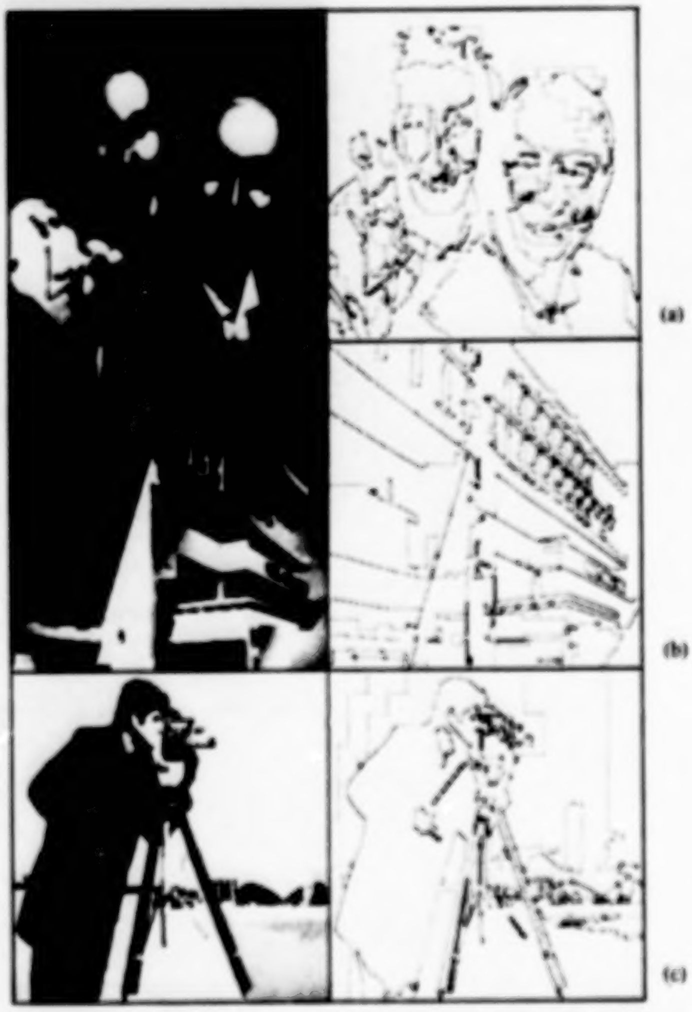


Fig. 5. Result of merge process. Each image is segmented into 49 regions. The compression ratios are 42:1 with third order polynomials (a), 53:1 with first order polynomials (b) and 68:1 with zero order polynomials.

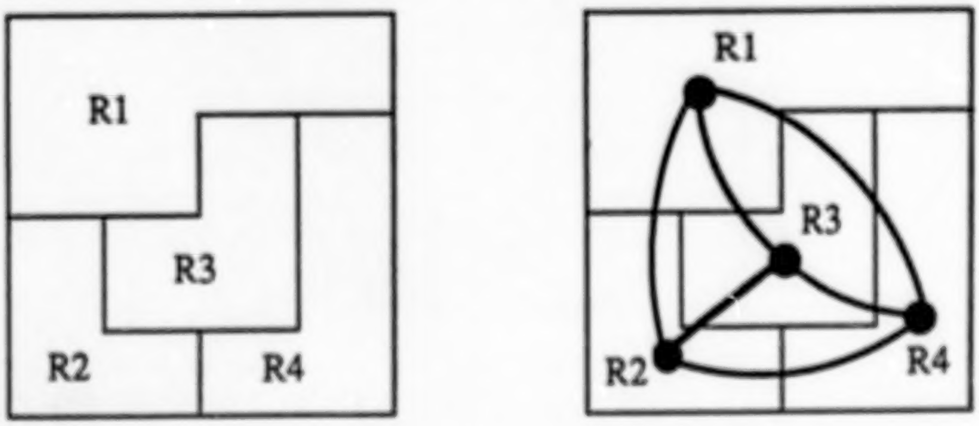


Fig. 6. Region space and its corresponding region adjacency graph (bidimensional representation)

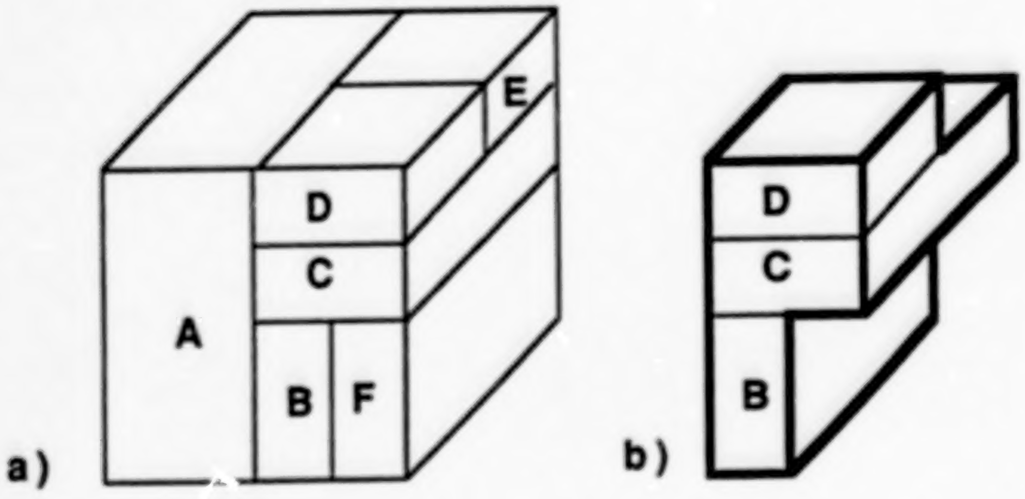


Fig. 7. a) Pyramidal structure defined in the entire data space.b) Region built from a set of parallelepipeds belonging to the pyramidal structure.

IMAGE PROCESSING BY INTENSITY-DEPENDENT SPREAD (IDS)†

Tom N. Cornsweet
University of California, Irvine

SUMMARY

As retinal illuminance is lowered, the human visual system integrates the effects of photon absorptions over larger areas and longer times. A theory of the process that might underlie these changes is called Intensity-Dependent Spread (IDS). Each input point gives rise to a pattern of excitation that spreads to a region of output points, each output point delivering a signal proportional to the total excitation it sees. The unique aspect of the theory is the assumption that, although the amplitude of the excitation pattern at its center increases with input illuminance, its width decreases in such a way that its volume remains constant.

Application of this theory to image processing reveals that it displays a number of unexpected and potentially useful properties. Among them are edge enhancement and independence from scene illumination.

INTRODUCTION

During the last several years, some of my colleagues and I have been working with an interesting image processing technique called Intensity-Dependent Spread, IDS (ref. 1,2). The things I will say here are the result of my working with Jack Yellott, Steve Reuman, and Greg Reese and of discussions with a lot of other people, George Westrom, Fred Huck, and Ellie Kurrasch to name a few. I would like to discuss some of the basic properties of IDS here; some of the papers that follow this one will present specific implementations and applications of the technique.

IDS was originally developed as a theory to explain some important phenomena in human brightness perception and it has turned out to do remarkably well at relating a number of phenomena that had always before been considered quite independent of each other. But it quickly became evident that the theory had potential as a useful computer image processing algorithm too and, although I will hint at some of the relevant phenomena of human vision in this paper, the following discussion will largely be confined to IDS as an image processing technique.

---

†This work was partially supported by NASA contract NAS1-18468.

SOME PRELIMINARIES

The IDS model, that is, the principles underlying IDS as an image processor, can be stated very simply. However, for those who are intimate with standard image processing techniques, there are some pitfalls to understanding it that need attention. In a typical image processing procedure, for example convolving an input image with a difference of Gaussians to achieve "edge enhancement", the value at each output pixel is determined by operating on a set of corresponding input pixels, in this example applying a weighing function and then adding up the results. Although one can correctly understand the IDS procedure in that same way, I think it is much easier to understand it and to avoid pitfalls if one imagines the process as we think of optical image formation. That is, each point in the input (scene) delivers its signal (light) to some region of the output (image). This spread of signal in the image plane from a unit point in object space is called the Point Spread Function (PSF). The image can then be considered the summation of the images of all the points in object space, that is, the convolution of the PSF and an ideal image.

Now I want to introduce a new term. The PSF is the distribution of light in the image of a point of unit intensity. If two point sources are imaged and one source is twice as intense as the other, although it can be said that the PSF's are the same, the actual distributions of light in the two images are different, one being twice as intense as the other at every image location. To talk about IDS, we need a term that permits that differentiation. Here I will refer to the actual distribution of signal that corresponds to a particular input point as the Signal Spread, SS. The SS and the PSF for a given input point are only the same when the point happens to have unit intensity.

In an ordinary image, the volume of the SS equals the total flux emitted by the corresponding object point multiplied by a constant representing the proportion of emitted light captured by the imaging system.

I will write that in the following peculiar way:

(1)

$$V = k * Q^1$$

where V is the volume of the SS,
Q is the number of quanta emitted by the point and
K is the proportion of light captured by the imaging system.

THE MODEL

The diagram at the top of Figure 1 schematizes the IDS model. The input is represented as a distribution of values in an array of input pixels. Each input pixel delivers a signal to a network, where the signal spreads laterally. Finally, there is an array of output pixels each of which simply sums all the signals that arrive in its vicinity from the network. In our new terminology, the SSs are developed in the network and the output array sums them. Further, in the version of the IDS model to be discussed here, the SS's are everywhere positive.

If the SS's were simply, say, Gaussians whose amplitudes were proportional to the corresponding input intensities, then this model would just describe ordinary linear low-pass filtering, for example, as would result from diffraction at the pupil of the imaging system. The linear model would be appropriate if all of the energy in the output distribution were to come directly from the input, as in an optical image, or perhaps with local amplification, as is true in a photographic image or, in effect, with a standard television image. The IDS model is based on a different physical notion, that the energy at each input point modulates the corresponding SS, and specifically, that the input does not determine the energy in the SS but instead affects the degree to which it spreads.

Now we can state the central feature of the IDS model. Although the mathematical form of the SS is constant, for example, it is always Gaussian or conical or cylindrical, etc., its width changes inversely with input intensity. Specifically, in IDS processing, the height at the center of the SS increases when the input intensity increases and its width decreases in such a way that the volume of the SS is constant. An example for an SS of conical shape at two input intensities is shown at the bottom of Figure 1. The following equation expresses this relationship.

$$(2) \qquad V = K*Q^0$$

(Equations (1) and (2) are written this way partly to clarify an important aspect of the relationship between IDS and a linear system, but it is also meant to suggest that it would be interesting to look at the consequences of using exponents other than 1 and 0.)

That is the entire IDS model. I will just fill out two details. First, the height at the center of the SS is taken as some power function of the corresponding input strength. The simplest such function, which will be used in the following examples, has a power of one, that is, the center height is linear with input intensity. Second, although the specific details of the results are somewhat affected by the particular spread shape chosen, e.g., Gaussian vs cylindrical, all of the general properties I will discuss here apply for spread functions of any shape.

GENERAL CHARACTERISTICS OF IDS-PROCESSED IMAGES

Applying the IDS model or process to images produces some results that are surprising. Figures 2a and 2b summarize a group of important characteristics of IDS processing. IDS is inherently a non-linear process (because the Point Spread Function varies with local intensity, superposition is not obeyed), but the curves in Figures 2a and 2b can be interpreted as close relatives of the MTF of a linear system. Here we will call these curves Contrast Sensitivity Functions (CSFs). Consider just one curve first, say the one labeled "10" in Figure 2a. This curve shows that, at a mean intensity of 10 arbitrary units, the system acts as a bandpass filter. Therefore, for a step input the output will be a spatial transient, as plotted in Figure 3. That is, IDS does what is often called "edge enhancement". This is surprising because IDS involves no subtraction. The PSF's are everywhere positive. (If the system were linear, low frequency attenuation could only be achieved with a PSF containing some negative regions.)

Figure 2a also shows that as the mean intensity of the input changes, the CSF changes, a result that can only occur in a non-linear system. When the mean intensity increases, the entire CSF shifts toward higher spatial frequencies. Specifically, when the center peak height of the SS is linear with input intensity, the CSF shifts (on a log frequency plot) in direct proportion to the square root of the mean intensity (2).

The consequences of this shift are interesting. What the system does is automatically adjust its smoothing and spatial resolution in accordance with local photon noise. Suppose, for example, that there is a region of an optical input image that has a low mean irradiance, so that quantal fluctuations in that region render the image noisier there than in another, brighter, region. The SS's in the dark region will be larger, the CSF's there will be shifted toward lower frequencies, and each output pixel will summate signals coming from a larger input region. That is, photon detections will be summated over a larger region of the image, causing increased averaging or smoothing there. That is a good property to have, because if a region of the image is noisy, it is not possible to achieve high scene resolution there anyway. High resolution in the processing system just reveals the noise, not the details of the scene.

If, on the other hand, a region of the image has a high irradiance so that the photon statistics support high scene resolution, the IDS process automatically delivers narrow PSFs there and thus achieves high resolution.

The curves in Figure 2a plot the behavior of IDS for deterministic inputs. When the input is an optical image and the Poisson statistics of photon-matter interactions are taken into consideration, the result is as plotted in Figure 2b. At extremely low intensities the IDS process acts as a linear low-pass filter. This low-pass behavior is not exactly a consequence of the model itself, but rather will occur only when the probabilistic aspects of the input are extreme, as with photon-limited detection of extremely low light level images, and I won't discuss it further here. (See ref. 3 for a complete discussion).

Figure 4 demonstrates this property of IDS graphically. Imagine a simple scene consisting only of two adjacent regions one with a reflectance of 10% and the other of 15%, the scene being illuminated and imaged. The jagged curve at the upper left in Figure 4a is a plot of the irradiance in the image of the scene when the scene illumination is 100,000 arbitrary units, and the upper right-hand curve is the resulting IDS output. The curve on the left, the input curve, is computed assuming that the illuminating light follows Poisson statistics, as all light does, and that the sensing system noise is negligible. Thus, the jaggedness in the left curve is the result of quantal fluctuations. Some of this noise is transmitted to the output image on the right.

The pair of curves in Figure 4b show what happens when the illumination on the scene is reduced by a factor of ten. The mean image irradiances on the two sides of the edge are reduced by a factor of ten (note that the vertical axis scale is magnified by ten relative to the upper left curve) and the effect of photon noise is relatively increased (by the square root of 10). The corresponding IDS output distribution is broader but not noisier. (Note that the vertical scale of the output signals is not increased. The fact that the amplitude of the edge response is not changed will be discussed below.)

Moving to the curves in Figure 4c, d, e and f each successive curve shows the result of another ten-fold decrease in scene irradiance. At the lowest irradiances, individual photon detections are noticeable. Although the S/N of the input images obviously increases with decreasing scene irradiance, the noisiness of the IDS output does not. In fact, the S/N remains exactly constant for the IDS outputs, as measured either by the ratio of the mean edge response amplitude to the RMS value of the output away from the edge, or by the variance in the location of the zero crossings[1]. Thus, the IDS process yields a constant S/N for images or regions of images whose local S/N ratio varies as a consequence of quantal fluctuations.

Note that no parameters of the model were adjusted between the curves in Figure 4. With regard to the output S/N, there is only one parameter to adjust, the width of the SS at some signal input intensity. This value determines the S/N that will appear at the output.

Figure 5 shows the IDS outputs to a series of step inputs similar to those in Figure 4 but where noise is negligible. The input steps are of increasing amplitude, and any linear system will give output responses that correspondingly increase in amplitude. However, the ratios of values on the two sides of all the input steps in this figure are equal, 2:1, and the figure illustrates another important property of IDS for step inputs. The response amplitude depends exclusively on the ratio of the values across the input step.

Now imagine that the input patterns in Figure 5 are actually plots of the intensity distributions in the images of a step between two areas, one having twice the reflectance of the other, the different plots corresponding to different scene illuminations (as in Figure 4). It is then clear that, when the amplitudes of the edge responses are considered, the IDS responses to edges in a scene are independent of the level and the uniformity of the illumination on the scene. They depend only on the relationships among the reflectances in the scene. This property, independence from scene illumination, can be extremely useful when the physical properties of the surfaces in the scene are of interest. In perhaps the most important application of this property, we can show theoretically that the spectral reflectances, or more loosely the "actual colors", of objects in a scene can be determined regardless of the color of the illuminant, by applying IDS processing to each of a set of multispectral images. We are currently working on ways to exploit this IDS property in processing actual multispectral images.

A FEW SPECIFIC EXAMPLES OF IDS PROCESSING

Figure 6 illustrates the action of IDS on a television image. Because edges produce responses of equal magnitude whether in direct light or deep shadow, the output image has a much larger visual dynamic range than the unprocessed image.

An extreme case is shown in Figure 7. The input is a standard television image of a simulated space scene, using a model spacecraft and astronaut and simulating the intense shadows of space by careful baffling of the illumination. A disadvantage of IDS processing, the broadening of edge responses at low light levels, is also clearly illustrated here. Other examples of IDS processing will be given in other papers in this collection.

A MODIFICATION TO PERFORM TEMPORAL PROCESSING

In the IDS model, signals spread laterally from each input point. Suppose we add the postulate that this signal spread is not instantaneous, but rather that the signals propagate laterally with a constant velocity, as they might if they were carried by neurons, for

---

[1] These are not really zero-crossings but "base level" crossings, the base level being non-zero, dependent upon an arbitrary choice of a particular parameter of the model, and not important.

example. If the propagation velocity is taken to be very high compared with the processing rate, then all of the resulting outputs are as described above. Similarly, if the input image is stationary and the output is displayed only after the system has reached equilibrium, the results will be as above. However, if it is assumed that the lateral spreading of the signal occurs within a time scale of the same order as the time to process an image, then an interesting set of temporal properties are manifested.

Figure 8 plots temporal responses of the system when the input is a step change in the irradiance of a spatially uniform field. The different curves are for different step amplitudes. The curves suggest temporal band-pass filtering and show that, when propagation velocity is included, the temporal properties of IDS are closely analogous to its spatial properties, these curves being the temporal analog of the spatial edge responses in Figure 3. In fact, a plot of the response of the system to inputs of zero spatial frequency (spatially uniform fields) that are modulated temporally at various frequencies and with various mean irradiances looks very much like the corresponding spatial result shown in Figure 2. The system is a temporal band-pass filter that shifts toward higher frequencies as the mean irradiance increases. Thus, merely by adding the assumption that the signal spread occurs over time, the system then not only trades spatial resolution against spatial smoothing but also trades temporal resolution for temporal smoothing. That is, as light levels are reduced, the signals are automatically integrated over both larger areas and longer times.

CONCLUDING REMARKS

Certain properties of the human visual system change as the mean light level changes. In particular, as the light level is reduced, the human visual system sums the effects of detected photons over larger areas of the retina and over longer time intervals. The usefulness of that behavior in a quantum-limited detection system like the eye is clear. High system resolution in both the spatial and temporal domains is obviously useful at high light levels, but it is useless at low light levels because fine spatial and temporal detail are obscured at the input by photon statistics. To maintain a constant ability to detect an object over variations in illumination level, one must integrate over larger temporal or spatial regions, or both, as the illumination is lowered.

Intensity-Dependent Spread is an algorithm that automatically adjusts its spatial and temporal areas of integration in inverse relation to the local image irradiance in such a way that, for quantum limited detection, the S/N is constant and independent of image irradiance. The same algorithm also results in band-pass filtering and edge "enhancement", and produces responses to edges whose amplitudes are proportional to the ratios of irradiances on the two sides of the edge. It thus yields an output image of a scene that is relatively independent of the intensity and uniformity of the light illuminating the scene.

References

1. Cornsweet, T.N. "Prentice Award Lecture: A simple retinal mechanism that has complex and profound effects on perception", Amer. J. Optom. & Physiol. Optics, 62, 427 (1985).

2. Cornsweet, T. N. and Yellott, J. I. Jr., "Intensity-Dependent Spatial Summation", J. Opt. Soc. Amer. A, 2, 1769 (1985).

3. Yellott, J. I. Jr. "Photon Noise and Constant Volume Operators", J. Opt. Soc. Amer. A, 4, 2418 (1987).

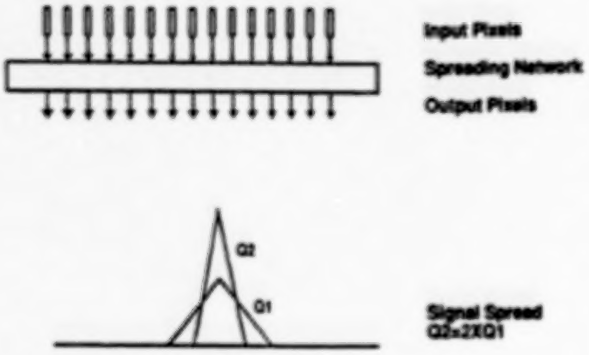

Input Pixels
Spreading Network
Output Pixels
Q2
Q1
Signal Spread
Q2=2XQ1

Figure 1 A schematic representation of the components of the IDS theory.

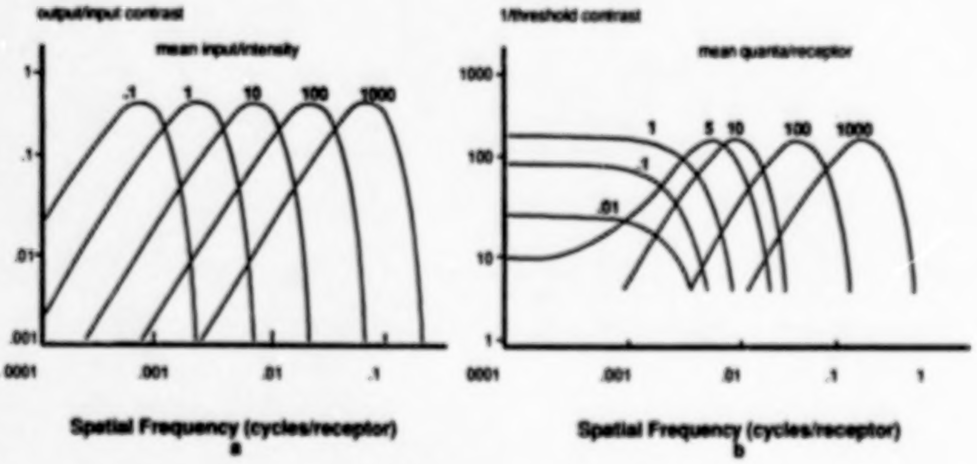


Figure 2 (a) Contrast sensitivity functions for IDS at various mean irradiances, assuming deterministic inputs. (b) Contrast sensitivity functions for IDS when photon statistics are included in the simulation. The lowest mean irradiances are such that the probability that a pixel will detect zero photons is significantly greater than zero.

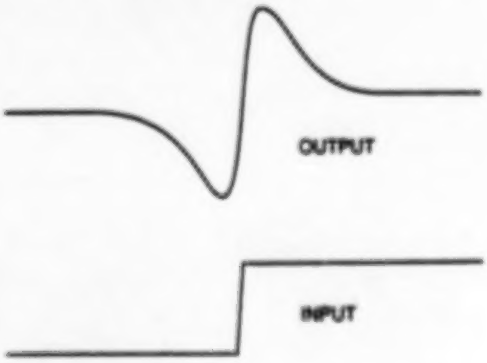


Figure 3 The IDS response to a step or edge.

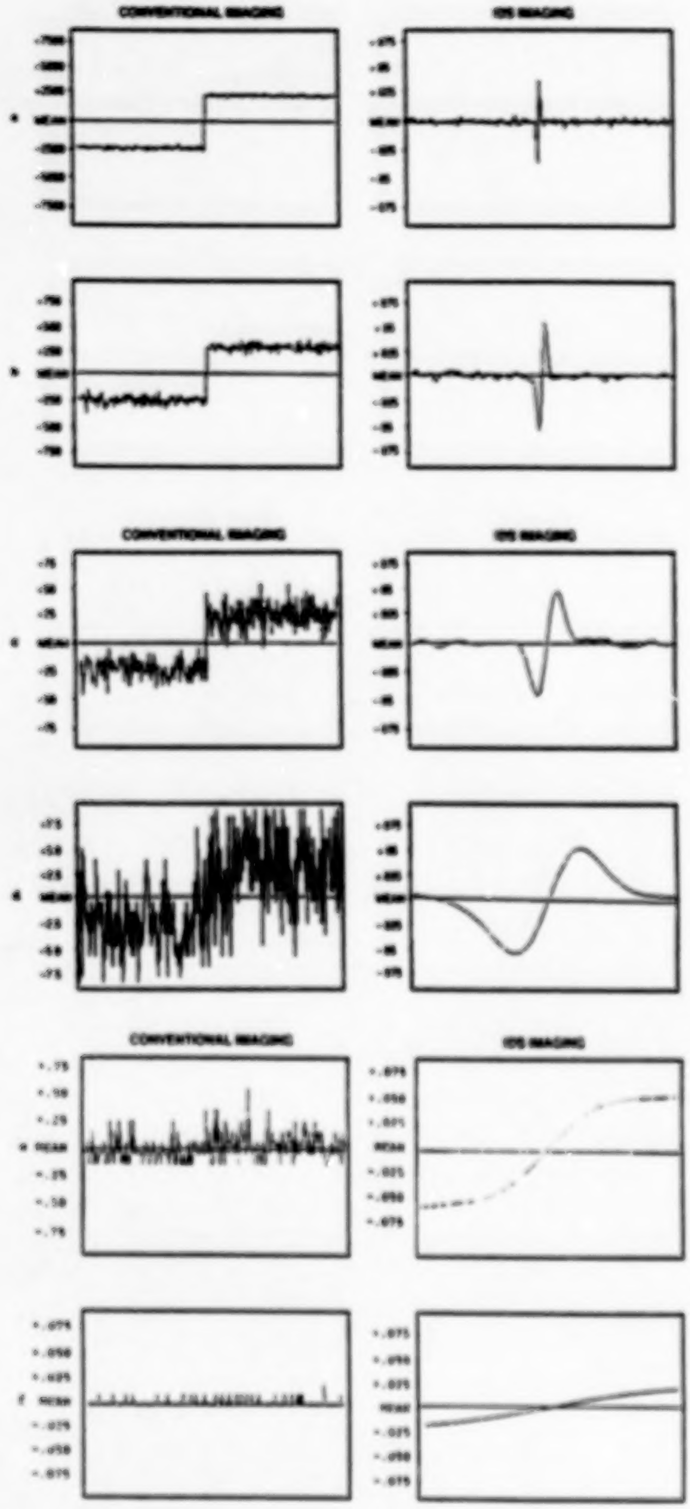


Figure 4 Each curve on the left is a plot of the relative irradiance in the image of a scene. The scene consists of two regions having reflectances of 10% and 15%. The image irradiances are computed on the basis of Poisson statistics. The curves on the right are the corresponding IDS responses. In (a), the scene irradiance is assumed to be 10,000 arbitrary units, and it is reduced by a factor of ten for each successive pair of curves.

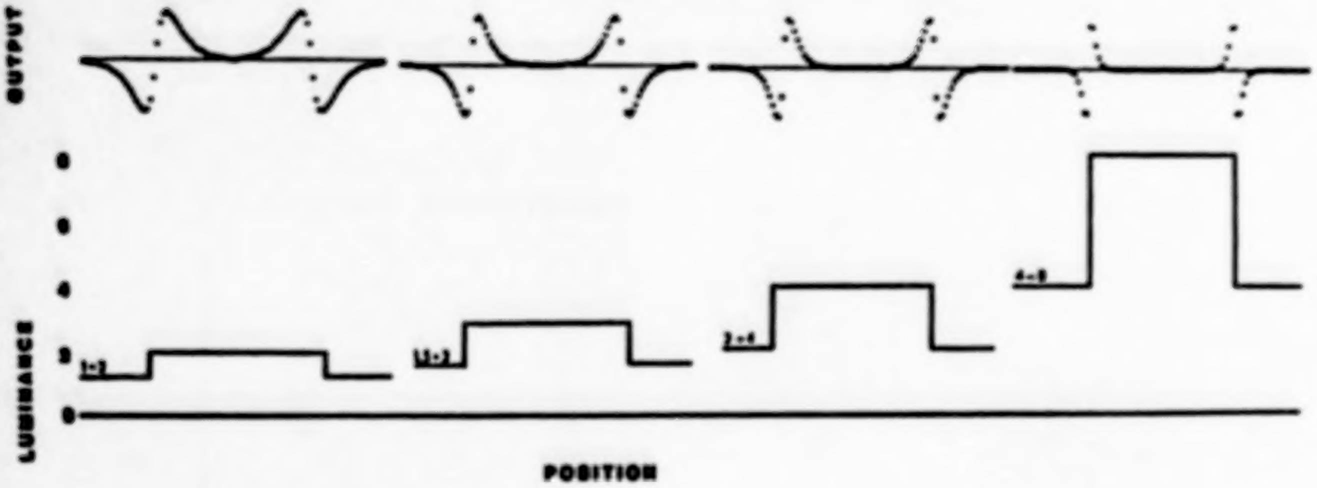


Figure 5 IDS responses to a set of deterministic step inputs, the ratio of image irradiances across the step being 2:1 in all cases.

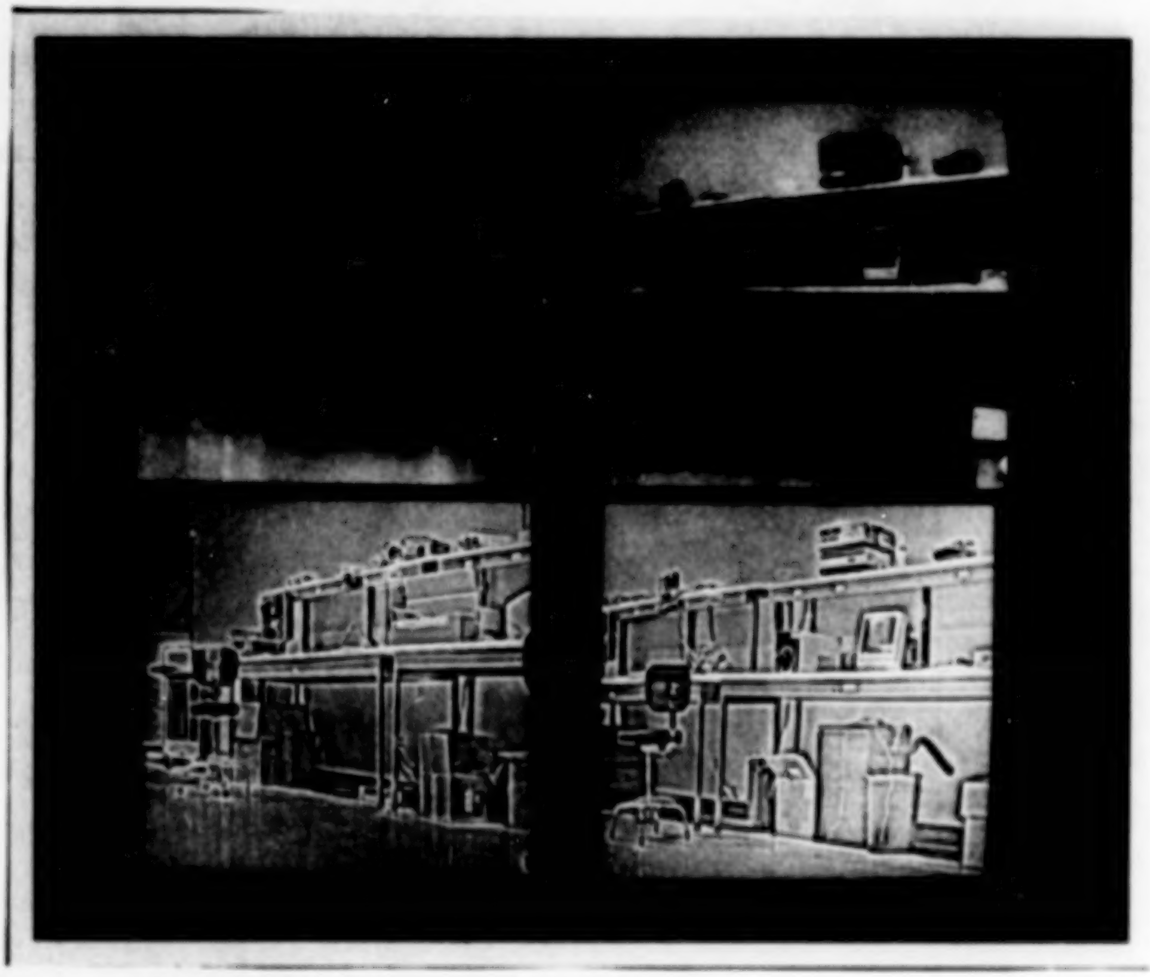
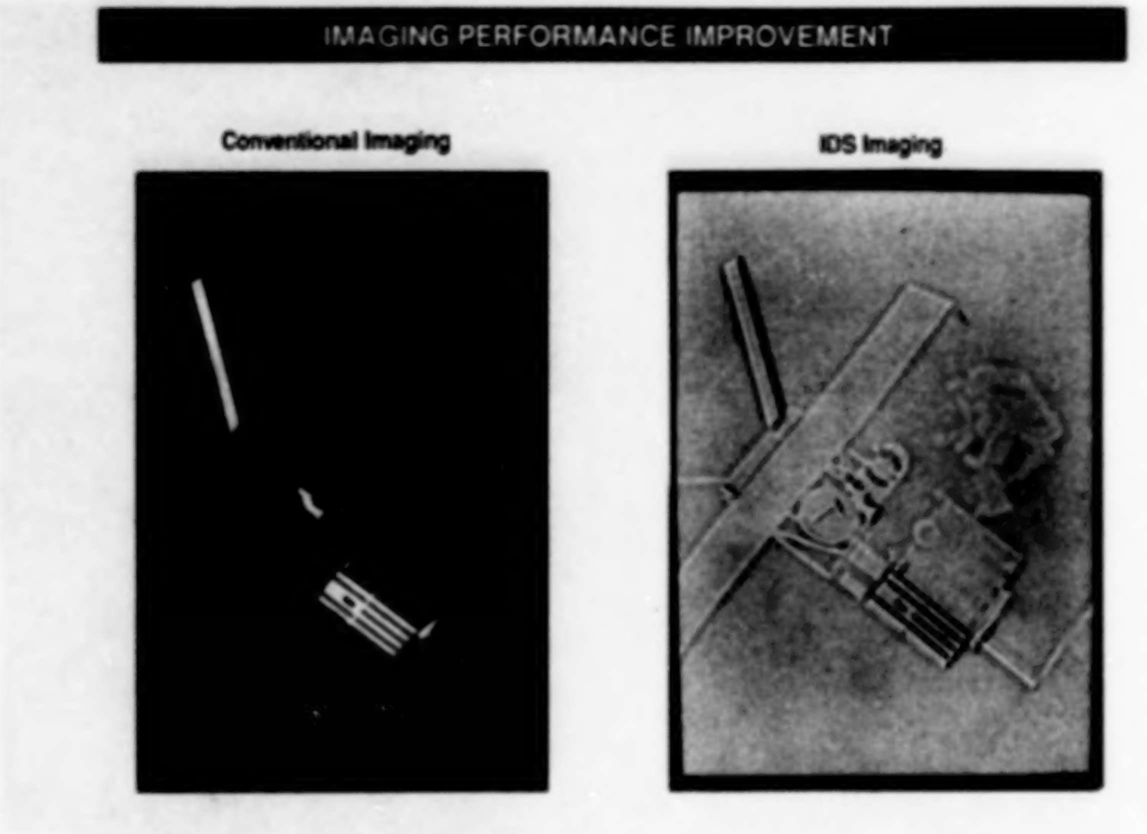


Figure 6 Two images captured by a standard television camera before and after IDS processing.



IMAGING PERFORMANCE IMPROVEMENT

Conventional Imaging

IDS Imaging

Figure 7 An image from a standard television camera of a scene simulating deep shadows in space and the result of IDS processing.

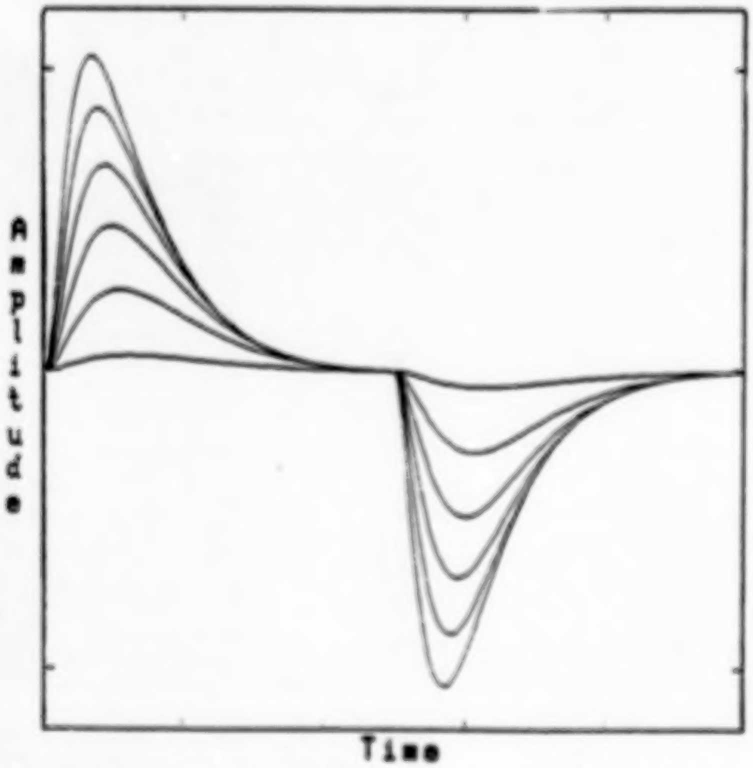


Figure 8 The responses of IDS to a spatially uniform field undergoing temporal step changes in irradiance of various amplitudes.

FROM PRIMAL SKETCHES TO THE RECOVERY OF INTENSITY AND REFLECTANCE REPRESENTATIONS

Rachel Alter-Gartenberg*
Old Dominion University, Norfolk, Virginia

Ramkumar Narayanswamy
Science and Technology Corporation, Hampton, Virginia

and

Karen S. Nolker
Computer Sciences Corporation, Hampton, Virginia

ABSTRACT

A local change in intensity (edge) is a characteristic that is preserved when an image is filtered through a bandpass filter. Primal sketch representations of images, using the bandpass-filtered data, have become a common process since Marr proposed his model for early human vision. In this paper, we move beyond the primal sketch extraction to the recovery of intensity and reflectance representations using only the bandpass-filtered data.

Assessing the response of an ideal step edge to the Laplacian of Gaussian ($\nabla^2 G$) filter, we have found that the resulting filtered data preserves the original change of intensity that created the edge in addition to the edge location. Using the filtered data, we can construct the primal sketches and recover the original (relative) intensity levels between the boundaries. Similarly, we found that the result of filtering an ideal step edge with the Intensity-Dependent Spatial Summation (IDS) filter preserves the actual intensity on both sides of the edge, in addition to the edge location. The IDS filter also preserves the reflectance ratio at the edge location. Therefore, we can recover the intensity levels between the edge boundaries as well as the (relative) reflectance representation. The recovery of the reflectance representation is of special interest as it erases shadowing degradations and other dependencies on temporal illumination.

This method offers a new approach to low-level vision processing as well as to high data-compression coding. High compression can be gained by transmitting only the information associated with the edge location (edge primitives) that is necessary for the recovery process.

1. INTRODUCTION

Primal sketches have become an important method of image description for low-level vision. One approach commonly used to produce these sketches is to bandpass filter the image data and then use the antisymmetrical signals created around intensity transitions (edges) to find their boundary location. We call the antisymmetrical signal a Mach-band pattern because it resembles the visual perception of an edge known as Mach-bands.[1] In this paper, we show that the Mach-band patterns contain more information about the original target than just the edge location. This additional information allows us to move beyond the extraction of primal sketches to the recovery of intensity and reflectance representations. Figure 1 demonstrates the recovery process of the intensity representation from the bandpass-filtered data for both a computer-generated target and a sampled image (e.g., image that is degraded by aliasing, blurring, and noise). The bandpassed images (b) of the targets (a) exhibit the familiar Mach-band patterns around the intensity transitions. Using the information contained in the Mach-band patterns we can recover the locations of the intensity transitions and extract the primal sketches (c) together with the actual change of intensity there. The recovery process uses this additional information to recover the original target (a), using only the bandpassed data (b) as is illustrated in (d).

---

* Mailing address: NASA Langley Research Center, MS 473, Hampton, VA

The conditions that allow us to recover a signal from partial information and specifically, the relationship between signals and their zero crossings, have been of considerable interest in the past. Logan[2] has set the conditions under which one-dimensional bandpass signals are uniquely specified by their zero crossings. Curtis et al.[3] set the conditions under which real, continuous, periodic, band-limited two-dimensional signals are specified from the zero-crossing locations of the real part of their Fourier transform. They applied their results to recover simple images from their threshold crossings. Independently, Yuille and Poggio,[4] and Hummel[5] showed that in the absence of image-gathering degradations, a target could be uniquely recovered from the information contained in its second derivative.

In this paper we assess the response of an ideal step edge to two models for retinal processing in human vision. We have found that the Mach-band pattern that results from filtering an ideal step edge with the $\nabla^2 G$ filter[6] preserves the original change of intensity that created the edge, in addition to the edge location (the zero-crossing location). Therefore, we can construct the primal sketches and recover the (relative) intensity levels between the boundaries. Similarly, we have found that the Mach-band pattern that results from filtering an ideal step edge with the IDS filter[7] preserves the actual intensity on both sides of the edge in addition to the edge location (the one crossing location). This filter also preserves the reflectance ratio (Weber fraction) at the edge location. Therefore, we can recover the intensity levels between the edge boundaries as well as the (relative) reflectance representation.

Our recovery method is local and uses the information contained in each edge element explicitly, recursively and independently. Therefore, this recovery process is quick and practical, with no need for extra memory, other than the storage for the recovered image itself, nor any extra calculations or processing.

2. PRIMITIVE EXTRACTION FROM THE LAPLACIAN OF GAUSSIAN FILTER

A. The $\nabla^2 G$ Filter

The model of lateral inhibition in early human vision processing and the assumption of white stationary Gaussian noise as a model for the natural noise source motivated Marr and Hildreth[6] to develop the spatially invariant $\nabla^2 G$ operator

$$\tau(x,y;\sigma) = \frac{1}{\pi\sigma^4}\left(1 - \frac{r^2}{2\sigma^2}\right)\exp\left(-\frac{r^2}{2\sigma^2}\right), \tag{1}$$

where $r^2 = x^2 + y^2$ and σ is the standard deviation of a normal distribution (Fig. 2). It is convenient to normalize the spatial variables relative to the sampling interval. Thus, for $\sigma = 1$, the standard deviation of the Gaussian is equal to the sampling interval.

The Gaussian is the only filter that guarantees a nice scaling behavior of the zero and level crossings of the linear differential operators.[4] It also is localized simultaneously and optimally in the spatial and spatial-frequency domains. The $\nabla^2 G$ operator is a linear isotropic bandpass filter that inherently satisfies the proper sequence of smoothing and differentiating for ill-posed differentiation problems (i.e., differentiating noisy data), and it assures smooth and stable zero-crossing curves.[8]

B. Step Edge Response

Isotropic filters allow a one-dimensional change of intensity from I to $I+\Delta I$ to simulate an ideal step edge. The response of this edge to the $\nabla^2 G$ filter, as given by

$$S_\sigma(x;\sigma,\Delta I) = \frac{\Delta I \cdot x}{\sqrt{2\pi}\sigma^3} \exp\left(-\frac{x^2}{2\sigma^2}\right), \tag{2}$$

is the familiar Mach-band pattern around the edge boundary that crosses zero exactly at the location of the change of the intensity. The corresponding peak and trough are symmetrically located at $x = \sigma$ and $x = -\sigma$ taking the values of

$$S_\sigma(\pm\sigma) = \pm\frac{\Delta I}{\sqrt{2\pi e}\sigma^2}. \tag{3}$$

The corresponding change of intensity becomes

$$\Delta I = \frac{\sqrt{2\pi e}\sigma^2[S_\sigma(\sigma) - S_\sigma(-\sigma)]}{2}. \tag{4}$$

Consequently, we have shown that the bandpassed data preserves the edge location and the change of intensity across it for spatial details that are at least 3σ wide (ideal case). Thus, when the bandpassed data $S_\sigma(x,y)$ is the only information available, the change of intensity associated with each edge element may become part of the image primitives in the low-level processing. The low-level processing consists of (1) detecting the zero-crossing location (x_o, y_o), (2) estimating the local edge direction θ, (3) measuring the values of $S_\sigma(x,y)$ at the points $(x_o \pm \sigma\sin\theta,\ y_o \pm \sigma\cos\theta)$, and (4) recovering ΔI using Eq. (4). The primitives (x_o, y_o), θ, and ΔI are used later to recover the (relative) intensity of the original image.

For the nonideal case, where a detail is insufficiently coarse relative to the spread of the filter, or where two edges cross each other or form a corner, it may often be possible to approximate ΔI. This approximation is achieved by measuring $S_\sigma(x,y)$ at the points $(x_o \pm r\sin\theta,\ y_o \pm r\cos\theta)$ where $r \leq \sigma$. Eq. (4) then takes on the form

$$\Delta I = \frac{\sqrt{2\pi}\sigma^3[S_\sigma(r) - S_\sigma(-r)]}{2r\exp\left(-\frac{r^2}{2\sigma^2}\right)} \tag{5a}$$

when $S(r)$ and $S(-r)$ are available and are approximately antisymmetrical and

$$\Delta I = \frac{\sqrt{2\pi}\sigma^3 S_\sigma(r)}{r\exp\left(-\frac{r^2}{2\sigma^2}\right)} \quad \text{or} \quad \Delta I = \frac{\sqrt{2\pi}\sigma^3 S_\sigma(-r)}{-r\exp\left(-\frac{r^2}{2\sigma^2}\right)} \tag{5b}$$

when only $S_\sigma(r)$ or $S_\sigma(-r)$ can be reliably measured.

C. Normalized Response

The Mach-band patterns at different scales (different σ) differ from each other not only by their spreads but by their amplitudes as well. The bandpass filter can be normalized in such a way that the response of an ideal step edge to the $\nabla^2 G$ filters at different scales will have the same amplitude and will differ only in their spreads. The scale S_σ for which the amplitude of the Mach-band pattern is exactly ΔI [i.e., $\Delta I = S(S_\sigma) - S(-S_\sigma)$] can be obtained from Eq. (4) as

$$S_\sigma = \left(\frac{2}{\pi e}\right)^{1/4} \cong 0.69. \tag{6}$$

To maintain constant amplitude ΔI , regardless of the operator size (i.e., the choice of S_σ), the bandpassed image should be multiplied by the normalization factor σ^2/σ_0^2. Figure 3 shows the response of the different operator sizes to an edge after normalization. The amplitude of the Mach-band pattern is exactly ΔI for all the responses. The only difference between them becomes the distance between the peak and trough and the zero crossings.

Consequently, for the ideal case, we can extract the change of intensity ΔI from the normalized bandpassed data, S, by the relationship

$$\Delta I = S(\sigma) - S(-\sigma). \tag{7}$$

Similarly, for the nonideal case, we can extract ΔI from the relationship

$$\Delta I = \frac{2}{b} S(r)\exp\left(\frac{1-b^2}{2b^2}\right), \tag{8}$$

where $b = \sigma/r$, and $r = \sigma$ refers to the ideal case.

The recovery of edge primitives obtained for the continuous and normalized response of an ideal edge to the $\nabla^2 G$ filter is in practice constrained by the sampling interval. This constraint is part of the imaging system and the inevitable transformation from continuous targets to their corresponding digital representations. It is interesting to observe that the scale σ_o corresponds to the size of the smallest scale with which edges can be reliably detected relative to the sampling interval of the image-gathering system. Therefore, σ_o determines the resolution of the recovery process. According to Marr et al.,[9] the smallest operator has a standard deviation $\sigma = 0.69$ relative to a normalized sampling interval. Furthermore, Huck et al.[10–12] have demonstrated that the resultant trade-off between aliasing and blurring in the image gathering for this response maximizes the acquired information density for high signal-to-noise ratios. The resolution of the recovery is constrained mostly by the sampling interval and not by the Gaussian blur represented by σ_o. In practice, it is preferable to let σ be 0.75 instead of 0.69. This slight increase in size appreciably reduces aliasing degradations for some signals.

D. The Recovery Process

The edge primitives extracted by the low-level processing just described, associated each edge point (x_o, y_o) with its local edge direction θ and its local change of intensity ΔI. In this section we present a method for recovering the (relative) intensity representation of the original image from these primitives. At first, we assign one of the regions of the image with an arbitrary initial intensity value I_o. From this region onward, we spread the values of $I + \Delta I$ and I toward the peak and trough of the Mach-band pattern, namely, in the $(x_o \pm r \sin\theta,\ y_o \pm r \cos\theta)$ directions. Spreads from different edge points toward the same region are averaged. Consequently, the image is recovered constructively, starting from one of the regions of the image. Each edge point joining the process provides a step towards the final representation of the recovered image. Theoretically, one estimate per region is sufficient to recover its (relative) intensity. However, we use estimates from all the ideal edge elements to attenuate the local error of each estimate. The recovery can be correct only up to a shift constant that is a function of the difference between the initial value I_o and the true intensity that corresponds to this starting region. The recovery process is quick and practical. It does not need high-order polynomial representations to describe the image,[3,4] nor does it need any extra calculations after the primitives (x_o, y_o), θ, and ΔI are obtained. The only memory needed for the recovery is that of the recovered image itself in which the edge elements are represented with their associate extra information θ and ΔI .

Our methodology extends the conditions for detecting stable edge curves from $\nabla^2 G$-bandpassed data set by Torre and Poggio.[8] They added to the zero detection from $S(x, y)$ the requirement that $|\nabla S(x, y)| \neq 0$ for the detected zero elements. Thus, a detector designed to extract zero crossings as edge elements from bandpassed images should also include information about the gradient of the bandpassed data near the detected zero crossings (usually referred to as slope). This additional information enables the detection of smooth and connected edge curves, defines corners in the image, and thresholds the noisy and disconnected elements from the true edge elements. In our low-level processing we changed the slope evaluation of the Mach-band pattern to its amplitude measurement. This slight change made the difference between the recovery of only the primal sketch description to the recovery of the intensities between these boundaries as well.

E. Accuracy and Stability

Inaccuracies in extracting the edge primitives introduced into the recovery process, even for an ideal case, are caused by the digital implementation of the (mathematically) continuous process. Further errors are introduced, in practice, by the image-gathering degradations (aliasing, blurring, and noise). It is necessary to choose the interval of processing (discretization interval) to be sufficiently small relative to both the sampling interval and σ, in order to minimize the inaccuracies of measuring $S(x, y; \sigma)$ and to assure stable estimates.

A recovery process that is based on estimates obtained from Eq. (7) or from Eq. (8) seems, initially at least, to be the same. However, estimating ΔI from Eq. (8) is like estimating it from $S'(x, y)$ (the gradient of $S(x, y)$), as opposed to the amplitude of $S(x, y)$, as we do in Eq. (7). The former estimation tends to be unstable, as was also observed by Hummel,[5] especially when the estimate is made for $r \ll \sigma$. The instability occurs because $S'(x, y) = \Delta I \cdot \nabla^2 G$ reaches its maximum at the edge location (x_o, y_o) which is near to where we measure $S(x, y)$ to approximate ΔI from Eq. (8) [Fig. 4(a)]. By contrast, $S'(x, y)$ is zero at $(x_o \pm \sigma \sin\theta,\ y_o \pm \sigma \cos\theta)$, where we measure $S(x, y)$ for the approximation obtained from Eq. (7). Small inaccuracies in measuring S near the edge location, where the gradient is the steepest, are amplified as a function of ΔI and result in large local errors when estimating ΔI (unstable recovery). On the other hand, small and local inaccuracies in measuring S near $r = \sigma$ are insignificant in the overall recovery (stable recovery).

The relative error in estimating ΔI from the Mach-band pattern is

$$\frac{\epsilon_{\Delta I}}{\Delta I} = \left[\left(1 - \frac{1}{b^2}\right)^2 + \frac{1}{\sigma^2} + \left(b + \frac{1}{b}\right)^2\right]^{1/2} \left(1 - \frac{1}{b}\right) \tag{9}$$

where $b = \sigma/r$, and r denotes the distance from the edge location to the location where S was measured. Fig. 4(b) illustrates the relative error $\epsilon_{\Delta I}/\Delta I$ as a function of $1 - 1/b$ and σ. As expected, the relative error is zero for $1 - 1/b = 0$ (i.e., at $r = \sigma$). Stable estimates can be obtained if the peak and trough of the Mach-band pattern are at a distance of at least three intervals of processing from the edge location (i.e., $3\Delta x = \sigma$ and Δx denotes the interval of processing, $\Delta x \leq 1$). Hence, the relative error is stable up to $1 - 1/b = \Delta x/2\sigma, \Delta x/2\sigma = 1/6$, (i.e., $\sigma - r \leq \Delta x/2$ and $3\Delta x = \sigma$), and diverges when $1 - 1/b > \Delta x/2\sigma$. A value of $\Delta x/2\sigma = 1/6$ results in an interval of processing that assures smooth discrete representation of the continuous $\nabla^2 G$. For such an interval of processing, the local relative error in estimating ΔI is smaller than 30% [see Figs. 4(c) and 4(d) for an actual σ], and can be controlled by averaging all the estimates for a given region.

Therefore, for a stable recovery, we recommend (1) implementation of filtering and processing with a discretization interval $\Delta x \leq 1$ that also obeys $\Delta x/2\sigma \leq 1/6$, $\sigma \geq \sigma_o$ and (2) the use of the estimates of ΔI obtained by Eq. (7). Eq. (8) may be used only when the actual peak or trough location falls between two discrete intervals of processing. After the entire image is recovered and a region remains with no estimate at all for its (relative) intensity, only then is it recommended to use the estimates of Eq. (8) to complete the recovery process (optional second stage of recovery). That way, the (relative) intensity of some small spatial details might be inaccurate, but the error will be local with no further propagation.

3. PRIMITIVE EXTRACTION FROM THE INTENSITY-DEPENDENT SPATIAL SUMMATION FILTER

A. The IDS Filter

Adaptive response to the intrinsic noisiness of light (photon noise) in early human vision processing motivated Cornsweet and Yellott[7] to develop the IDS filter. The IDS model consists of nonnegative, spatially homogeneous, circularly symmetric spread functions (SF) K, with unity volumes. The SF's differ from each other by their spreads, which are inversely dependent on the local intensity $I(x,y)$ as given by

$$\tau(x,y;I) = IK(Ir^2) \tag{10}$$

where $r^2 = x^2 + y^2$. Thus, the effect of the input intensity is to rescale the SF's, leaving their basic form unchanged [Fig. 5(a)]. The image response to the IDS model is the sum of the SF's [Fig. 5(b)]:

$$S(x,y;I) = \iint_{\infty}^{-\infty} \tau(x',y';I)dx'dy'. \tag{11}$$

The IDS operator is an isotropic spatially variant bandpass filter that exhibits the following properties:

(1) Its response to a nonzero uniform scene is unity

(2) Its response is invariant under translation and rotation

(3) Its response to an input intensity $cI(x,y)c > 0$ is

$$S[x,y;cI(x,y)] = S\left[x\sqrt{c},\ y\sqrt{c}; I\left(x/\sqrt{c},\ y/\sqrt{c}\right)\right]$$

That is, the height of the SF is increased by the factor c while its width is decreased by the factor $1/\sqrt{c}$ (scaling property).

In the discrete digital implementation, c is chosen so that the diameter of the SF for the highest intensity of the image overlaps with at least seven discrete image data. The distance between these data is then assumed to be $1/\sqrt{c}$. It can be interpreted as the physical separation between the sampled data (i.e., the sampling interval), or as the discretization of the continuous IDS model given by Eq. (11) (i.e., the interval of processing).

B. Step Edge Response

For the recovery purpose, we restrict ourselves to the family of feasible SF's for the IDS filter that are also separable. Similar to the response of an ideal step edge to the $\nabla^2 G$filter, the edge response to the IDS filter, as given by

$$S(x;I,\Delta I) = 1 + \int_0^{\sqrt{I+\Delta I}x} K(z)dz - \int_0^{\sqrt{I}x} K(z)dz = S\left(\sqrt{I}x;1,W\right), \qquad (12)$$

is a Mach-band pattern around the edge boundary. $W = \Delta I/I$ denotes the Weber fraction or the reflectance ratio. The Mach-band pattern crosses the value of one exactly at the edge location. The corresponding peak and trough are located symmetrically at a distance p that satisfies the equation

$$\sqrt{1+W}K\left(\sqrt{1+W}p'\right) = K(p') \qquad (13)$$

where $p' = \sqrt{I}p$, and they take on the value $S(p';1,W)$. The range of the amplitude of the Mach-band pattern is dictated by the unity volume condition of the SF's and is given by $1 - 0.5 < S(\sqrt{I}p;1,W) < 1 + 0.5$. The amplitude reaches its limits $1 > 0.5$ when $I \to \infty$. The IDS response has a constant amplitude for a constant reflectance ratio W, while its spread is also a function of the original intensities I and $I + \Delta I$ (Fig. 6).

The IDS-bandpassed data $S(x,y;I,\Delta I)$ retains information about the original image. Thus, the original intensities on both sides of the edge and the reflectance ratio associated with each edge element may become a part of the image primitives in the low-level processing. The reflectance ratio is recovered by measuring the amplitude of the Mach-band pattern and deriving W from Eq. (12). The original intensities are recovered by measuring the distance from the peak and trough locations to the one-crossing location. Substituting these distances and the estimation of W in Eq. (13) and solving it, we extract the I and $I + \Delta I$ primitives needed for the recovery.

C. Recovery From the Cylindrical IDS Response

The cylindrical function is a feasible SF for the IDS-recovery process. Its definition is

$$K(x,y) = \begin{cases} 1 & 0 \le \sqrt{x^2+y^2} \le 1/\sqrt{\pi} \\ 0 & \text{elsewhere} \end{cases}$$

while its corresponding line spread function is

$$K(x) = \begin{cases} 2\left(\frac{1}{\pi} - x^2\right)^{1/2} & 0 \le x \le 1/\sqrt{\pi} \\ 0 & \text{elsewhere} \end{cases}$$

We have chosen to analyze the IDS recovery process with the cylindrical SF for the following reasons:

(1) The cylindrical SF has a finite support. Therefore, the spread of the corresponding IDS operator is finite with a radius of $(\pi I)^{-1/2}$. Finite support assures accurate integration in the

discrete implementation of Eq. (12). SF's with an infinite support, such as the Gaussian, typically would require more than twice the processing to approach the same numerical accuracy, due to the larger support necessary for the integration. Accurate integration is mandatory to a stable recovery process, as we will show in subsection E.

(2) The primitives can be extracted explicity from the cylindrical-IDS Mach-band pattern, i.e., through a direct relationship between the Mach-band amplitude and the primitives. Primitives can be extracted only implicity from the Gaussian-IDS Mach-band pattern, i.e., through look-up tables.[7]

Substituting the cylindrical SF in Eq. (13), and solving it for p, we have the distance from the one crossing (edge location) to the peak location in the Mach-band pattern as

$$p = [\pi I(2+w)]^{-1/2}. \tag{14a}$$

The corresponding peak value derived from Eq. (12) takes on the value of

$$S(p) = 1 + \frac{1}{\pi}\sin^{-1}\left(\frac{W+1}{W+2}\right)^{1/2} - \frac{1}{\pi}\sin^{-1}\left(\frac{1}{W+2}\right)^{1/2} \tag{14b}$$

Consequently, the low-level processing consists of estimating the local edge direction θ at the one-crossing location (x_o, y_o), and measuring the peak and trough values $S(p)$ at their corresponding locations $(x_o \pm p\sin\theta, y_o \pm p\cos\theta)$. We can then recover the original ideal edge parameters (edge primitives) $W = \Delta I/I$, I, and ΔI, using the bandpass signal information (i.e., p and $S(p)$) from the relationships

$$W = \frac{2\sin\phi}{1-\sin\phi}, \quad I = [2p^2(W+2)]^{-1}, \quad \Delta I = WI \tag{15}$$

where $\phi = \pi[S(p) - 1]$ for the peak measurements and $\phi = \pi[1 - S(p)]$ for the trough measurements. The primitives (x_o, y_o), θ, W, and I are used later to recover the (relative) reflectance and the intensity representations of the original image.

D. The Recovery Process

The recovery process from the IDS-bandpassed data is similar to the recovery process described for the $\nabla^2 G$-bandpassed data. The initial low-level processing associated each edge point (x_o, y_o) with its local edge direction θ, local reflectance ratio W, and local intensities I and $I + \Delta I$. Spreading these primitives onto their corresponding region in a similar process described in Section 2.D would result in the original intensity representation and the (relative) reflectance representation. The latter representation can be correct only up to a constant factor that relates to the initial reflectance that started the spread processing.

E. Resolution Accuracy and Stability

Resolution

The discrete display elements of the input data $I(x_i, y_i)$ determine the discretization intervals of the discrete IDS-bandpassed output image $S(x_i, y_i)$. The scaling property of the IDS helps us understand the transformation between the continuous representation of the IDS model and the discrete representation of the image data. We choose the scaling c so the diameter of the smallest spread of the IDS operator (e.g., at the highest intensity) will overlap at least seven discrete image elements. Thus, the distance between two discrete elements Δx (smaller or equal to the sampling interval) has a physical distance of $\Delta x = 1/\sqrt{c}$. Therefore, when we measure the distance p in terms of intervals of processing, we should multiply it by $1/\sqrt{c}$ to be able to use Eq. (15) that was derived from the continuous representation of the IDS model. We do the same when we assess the resolution of the image recovery process.

The distance $p_{\Delta x}$ between the edge and peak locations in terms of intervals of processing is $[\pi \ \Delta x \ I(2+W)]^{-1/2}$. For images that are sufficiently sampled, features larger than $2p_{\Delta x}$ can be reliably recovered with the primal sketch description, along with the recovery of the (relative) reflectance or intensity representations. Spatial features smaller than $2p_{\Delta x}$ are blurred by the filter. Fig. 7 illustrates the resolution of the recovery for each intensity as a function of the scaling parameter c for a given reflectance ratio W.

Accuracy

The accuracy with which the reflectance ratio W can be recovered depends on the accuracy with which the peak and trough values of $S(x, y)$ are measured:

$$\epsilon_W = \frac{2\pi \cos\phi}{(1-\sin\phi)^2}\epsilon_s = \pi(W+2)\sqrt{W+1}\,\epsilon_s,$$

where $\phi = \pi[S(p) - 1]$. The peak and trough curves of the Mach-band pattern are parallel to the edge curve. Therefore, a small error in estimating the local edge direction θ, which determines the search direction for the peak and trough, does not affect the value $S(p)$. Moreover, if for an accurate θ, the actual peak or trough falls between two intervals of processing, then one of the neighboring elements is often a better measurement that can be obtained easily. Therefore, the error ϵ_s depends mostly on the quantization used in the digital implementation of the IDS (8 bits in our case) and is less than 30% for $0.2 < W < 128$ (on the [0,255] range), assuring a stable recovery [Fig. 8(a)].

The recovery of the original intensity, on the other hand, depends strongly on the accuracy with which the distance p is measured. The dual relationship between I and p as given by $\pi I p^2(2+W) = 1$, leads to the relative error ϵ_I/I in estimating I. For a measured distance p and an estimated W through a measurement of $S(p)$ the relative error is

$$\frac{\epsilon_I}{I} = \left[\pi^2(1+W)\epsilon_s^2 + 4\pi I(W+2)\epsilon_p^2\right]^{1/2},$$

where $\epsilon_p^2 \leq 1/4c$ [Figs. 8(b) to 8(d)]. The second term of ϵ_I/I is the dominant one. It can become high for orientations θ that do not agree with the rectangular lattice or parameters c that are not high enough. On the other hand, choosing a higher c would further blur the fine details so they would be unrecoverable. The scaling parameter c, which sets the limitation on the resolution of the recovery clearly demonstrates the usual trade-off between higher resolution (smaller c) and the accuracy of the recovery that can be obtained by increasing c [Figs. 7 and 8]. Accurate intensity recovery depends strongly on the type of the original scene and on the digital implementation of the IDS operator. Initial inaccuracy in the integration will further degrade the estimates of I due to errors in determining the one crossings and the distortion of the symmetry of the Mach-band pattern around it.

Stability

For each edge element that we recover from the IDS-bandpassed data we have two estimates for both the reflectance ratio W and the original intensity I [Eq. (15)]. One estimate is from measuring p and $S(p)$ at the peak of the Mach-band pattern, and the other one is from measuring p and $S(p)$ at the trough of the Mach-band pattern. We consider an estimate of W or I to be stable when the estimation through the peak and trough are about the same. The estimation that we use for the recovery process is the average of the two. Unstable estimates may occur when:

(1) The feature that we try to recover is smaller than $2p_{\Delta x}$

(2) Two Mach-band patterns interfere (corners, crossings, etc.)

(3) Edge orientation causes large inaccuracies in measuring p

(4) The reflectance ratio W is too high

For a stable recovery we go through a process that is similar to the $\nabla^2 G$recovery. We begin recovering the original image using only the stable estimates obtained from the IDS-bandpassed data. After the entire image is recovered and a region remains with no estimate at all for its reflectance or intensity (depends on which representation we wish to recover), only then do we use the unstable estimates to complete the recovery process. For images that are sufficiently sampled, regions with no stable estimate are minimal. While all we need is one stable estimate to recover a feature without using unstable estimates, we actually have stable estimates (that are averaged for smooth appearance) from most of the edge elements. The intensity recovery, although stable for images that are sufficiently sampled, tends to be less accurate than the (relative) reflectance recovery due to the inevitable discretization process.

4. RESULTS

In this section we characterize the images that are recovered from both the $\nabla^2 G$ and IDS-bandpassed signals without the use of any other data. The original image data are either computer-generated or obtained from a mock setup in space. Together these targets present a variety of different scene characteristics. These results combine the accuracy and stability assessments for a full recovery and the trade-offs for cases when only partial recovery can be obtained.

Figure 9 summarizes the recovery of image characteristics from the bandpassed data. The original targets are computer generated targets with features that are coarse enough relative to the sampling interval. The bull's-eye target that simulates staircase edges allows us to examine how constant change in the edge direction affects the quality of the recovery. The square target which is tilted 30° relative to the lattice allow us to examine how this tilt affects the recovery. The square and the random rectangles targets also allow us to examine how corners and crossings in a full range intensity image affect the recovery process. The spread of the $\nabla^2 G$ filter used herein was controlled by $\sigma = 1.0$ and the spread of the cylindrical IDS was controlled by $c = 6000$.

Using only the bandpassed data, the first stage in the recovery process is to extract the location of the intensity transitions (edges) through the zero crossing of the $\nabla^2 G$-bandpassed data or the one crossings of the IDS-bandpassed data. The resulting primal sketches are illustrated in Fig. 9(b). The recovery of the (relative) intensity representation from the $\nabla^2 G$-bandpassed data is illustrated in Fig. 9(c). The quality of the recovery is measured by the cross correlation ρ between the original target and the recovered one. The recovery of the intensities and the (relative) reflectance representations from the IDS-bandpassed data are illustrated in Figs. 9(d) and 9(e), respectively. As expected from the accuracy assessment, the quality of the (relative) reflectance recovery is better than the IDS intensity recovery. The high correlations between the original targets and the recovered ones suggest that the recovery process may be used as a decoder for a coding scheme in which only the edge primitives are transmitted.

Figure 10 illustrates a particularly important characteristic of the IDS filter, namely, the robustness of its reflectance representation to local variation in illumination (e.g., shadow). The recovered target [Fig. 10(d)] resembles the original [Fig. 10(a)], and not the shadowy [Fig. 10(b)], which is the one that was filtered. Traces of the shadow degradation can be seen in the modest loss of accuracy in the actual transition as the illumination decreases.

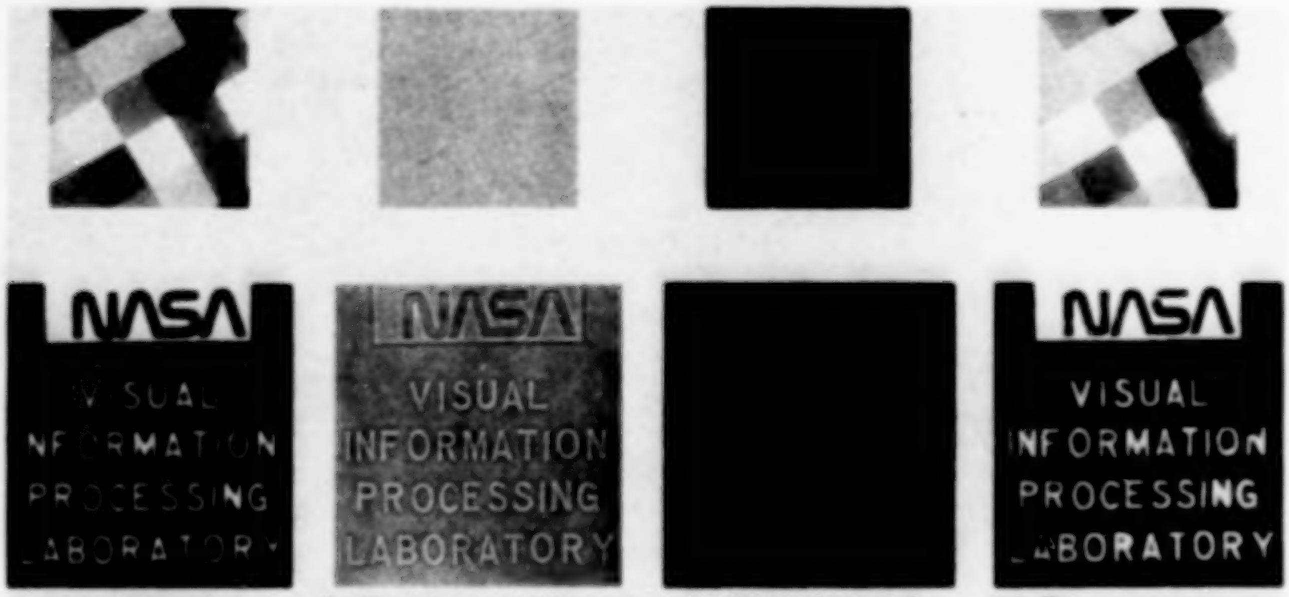
Figure 11 illustrates the capability of the recovery process with an experimental setup that simulated imaging conditions in space. The target examines the recoverability of targets with a wide dynamic range of intensities, in particular, the recoverability of spatial details under direct illumination or in deep shadow. As can be observed from the edge recovery representation [Fig. 11(b)], many important features of target, including features in deep shadow, could be recovered. The recoveries from the $\nabla^2 G$ (with $\sigma = 3.0$) are illustrated in Fig. 11(c), and those from the IDS (with $c = 6000$) are illustrated in Fig. 11(d) and Fig. 11(e). These recoveries resemble a slightly blurred version of the original target [Fig. 11(a)]. As could be expected from the accuracy and stability analysis, features that were relatively small produced inaccuracies as well as features with abrupt and steep changes in the intensities. Nevertheless, in most of the cases, these inaccuracies were contained within their regions with no further propagation.

5. CONCLUDING REMARKS

Assessment of the response of an ideal edge to a bandpass filter reveals that most of the target's characteristics are preserved. For applications that use bandpass signals, the recovery of those characteristics exhibits new dimensions to image understanding. Minimal extra processing is required, beyond that is needed for the primal sketch extraction process. The processing is based on the existing bandpassed data information and on the same storage needed for the recovered image itself. The potential for high data compression applications, transmitting only the information associated with detectable edge boundaries (edge primitives), might be helpful when high data rate transmission is required. In light of our results and the stability assessment, we feel that the edge primitives information extracted from the bandpassed image is a good form of representation for images that are sufficiently sampled and properly processed.

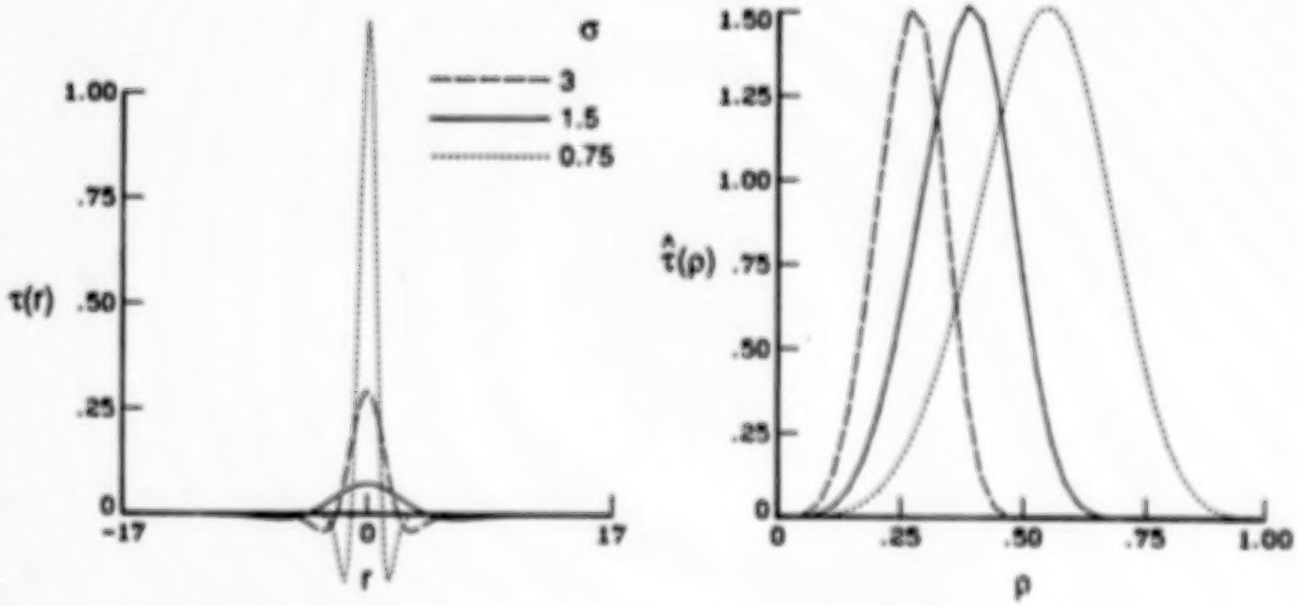
REFERENCES

1. T. N. Cornsweet, *Visual Perception*, Academic Press, New York, 1970, pp. 277 and 304.
2. B. F. Logan, "Information in the zero crossing of bandpass signals", The Bell System Technical Journal, Vol. 56, No. 11, pp. 487-510, 1977.
3. S. R. Curtis, A. V. Oppenheim, and J. S. Lim, "Signal reconstruction from Fourier transform sign information", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-33, No. 3, pp. 643-657, June 1985.
4. A. L. Yuille and T. Poggio, "Fingerprints theorems for zero crossings", J. Opt. Soc. Am., Vol. 2, No. 5, pp. 683-692, 1985.
5. R. A. Hummel, "Representations based on zero crossings in scale-space", IEEE Proc. of the Computer Vision and Pattern Recognition Conference, pp. 204-209, June 1986.
6. D. Marr and E. Hildreth, "Theory of edge detection", Proc. R. Soc., London B 207, pp. 187-217, 1980.
7. T. N. Cornsweet and J. I. Yellott, Jr., "Intensity-dependent spatial summation", J. Opt. Soc. Am., 2, pp. 1769-1786, 1985.
8. V. Torre and T. A. Poggio, "On edge detection", IEEE Trans. Pattern Anal. Machine Intell., PAMI-8, pp. 147-163, 1986.
9. D. Marr, T. A. Poggio, and E. Hildreth, "Smallest channel in early human vision", J. Opt. Soc. Am., Vol. 70, No. 7, pp. 868-870, July 1980.
10. F. O. Huck, C. L. Fales, D. J. Jobson, S. K. Park, and R. W. Samms, "Image-plane processing of visual information", Applied Optics, Vol. 23, No. 18, pp. 3160-3167, 1984.
11. F. O. Huck, C. L. Fales, N. Halyo, R. W. Samms, and K. Stacey, "Image gathering and processing: Information and fidelity", J. Opt. Am. A2, pp. 1644-1666, 1985.
12. F. O. Huck, C. L. Fales, J. A. McCormick, and S. K. Park, "Image-gathering system design for information and fidelity", J. Opt. Soc. Am. A5, pp. 285-299, 1988.


NASA
VISUAL
INFORMATION
PROCESSING
LABORATORY

(a) Target (b) Bandpassed image (c) Primal sketch (d) Recovered target

Figure 1: The recovery process from the bandpassed representation to the original image representation.



(a) Spatial domain (b) Frequency domain

Figure 2: Normalized $\nabla^2 G$ response for three standard deviations σ.

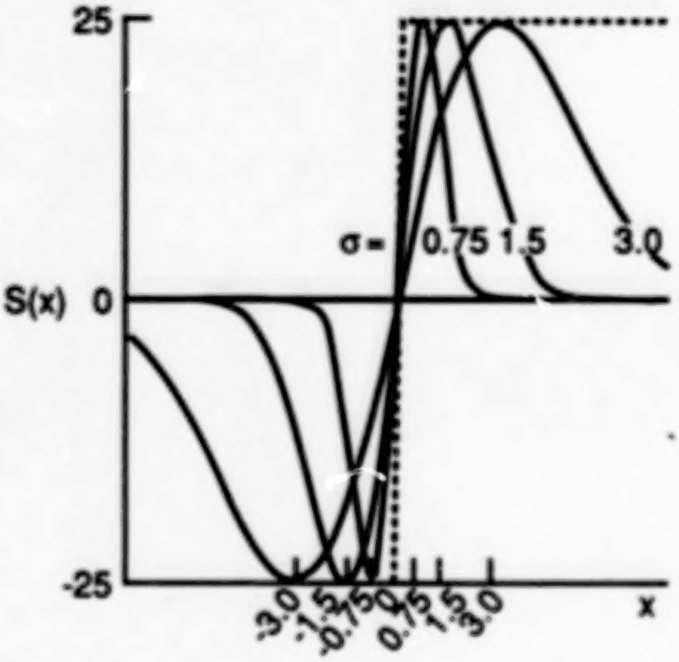
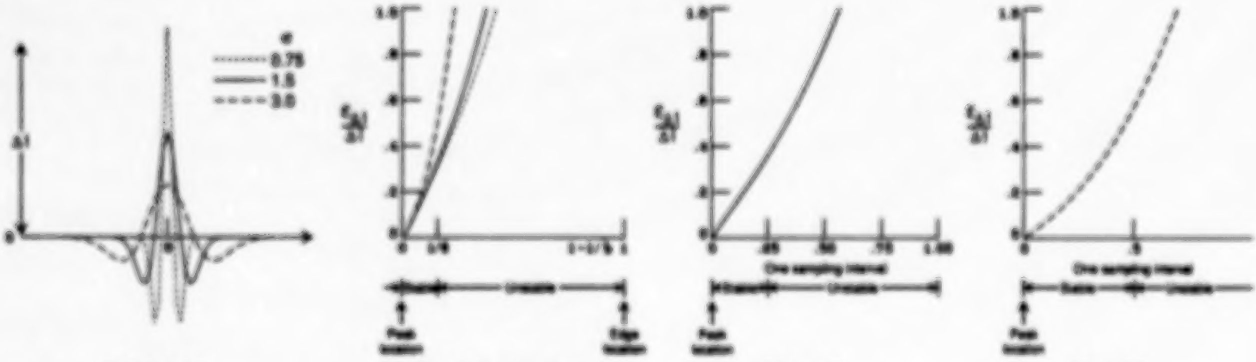
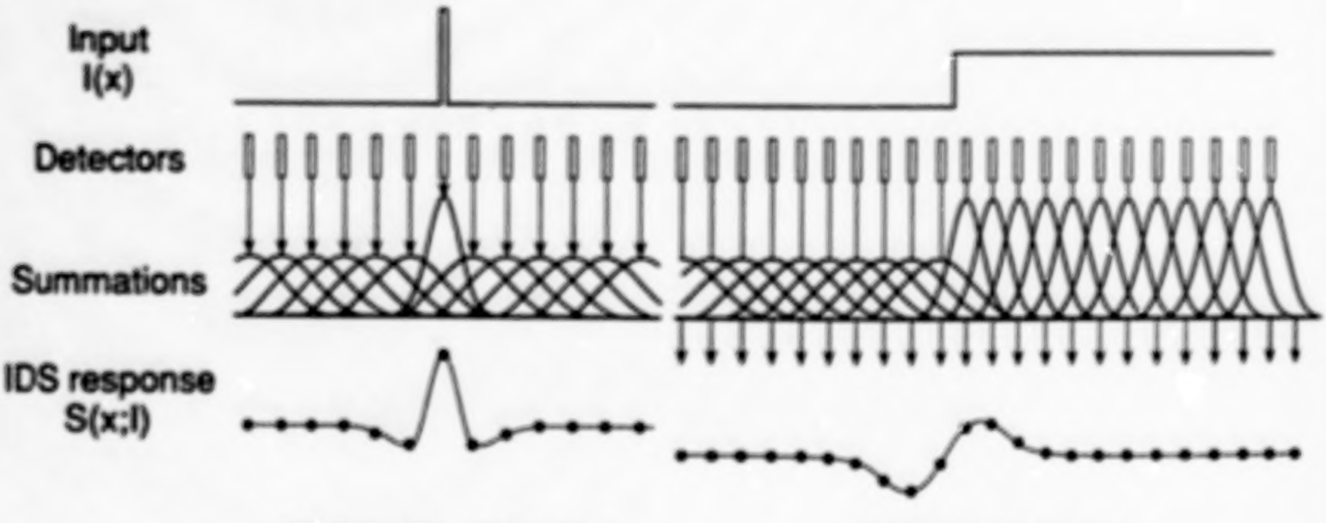


Figure 3: Normalized response to an ideal step edge for $\nabla^2 G$ operators with different σ.



(a) $\tilde{\nabla}(x;\sigma,\Delta I)$ (b) The relative error $\varepsilon_{\Delta I}/\Delta I$ (c) $\sigma = 1.5$ (d) $\sigma = 3.0$

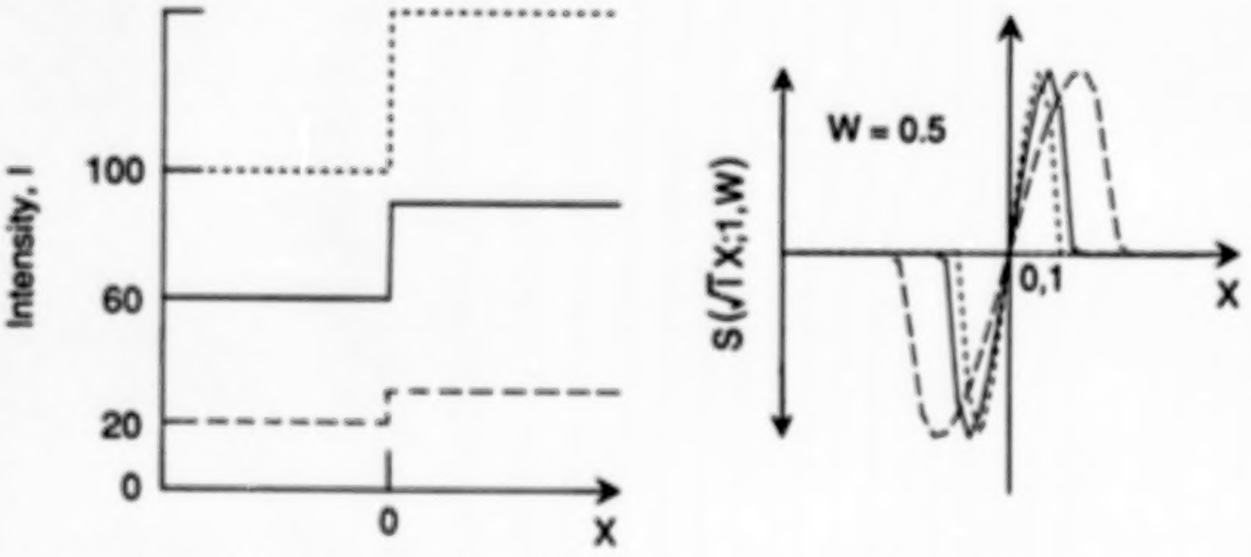
Figure 4: Accuracy of the $\nabla^2 G$ recovery process.


Input
I(x)
Detectors
Summations
IDS response
S(x;I)

(a) Impulse response

(b) Edge response

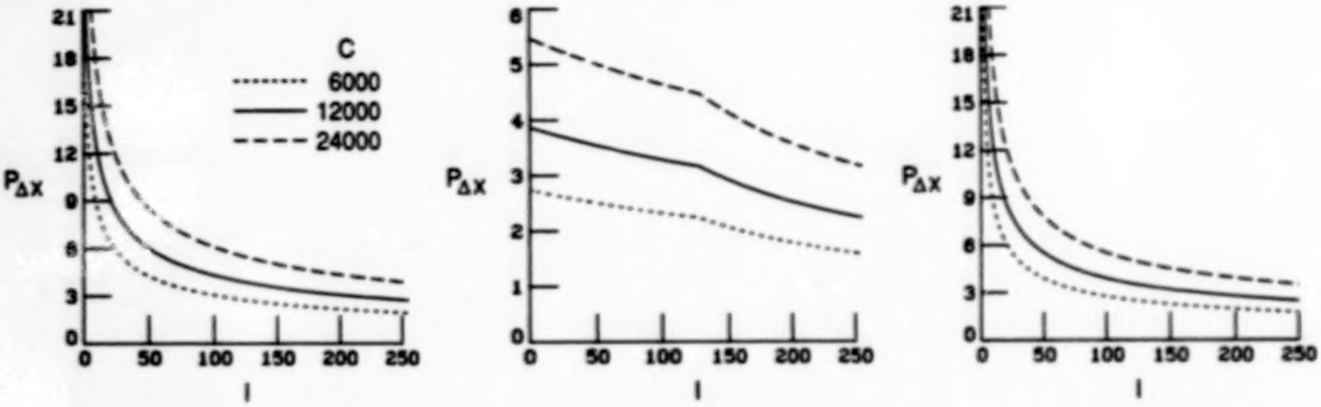
Figure 5: The IDS response.



(a) Edges with a constant W = 0.5

(b) IDS response

Figure 6: Response of ideal step edges to the IDS operator.

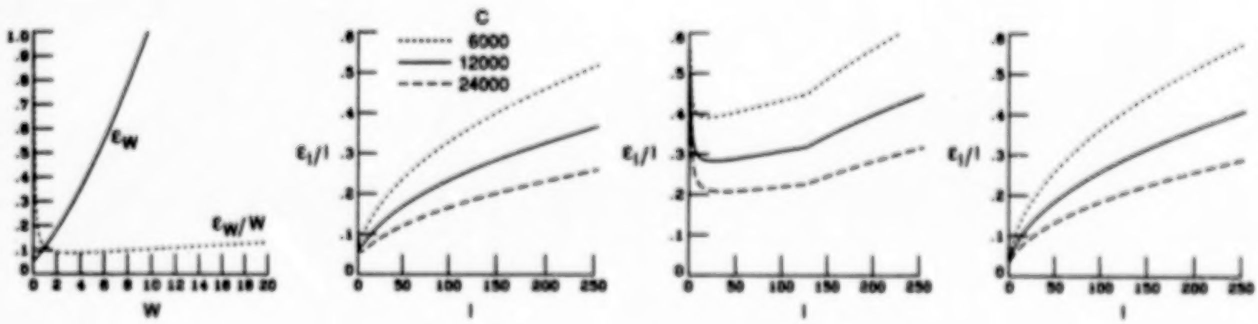


(a) W = 5/I

(b) W = max $\left[\frac{255-I}{I}, 1\right]$

(c) W = 0.5

Figure 7: Resolution of the IDS recovery process.



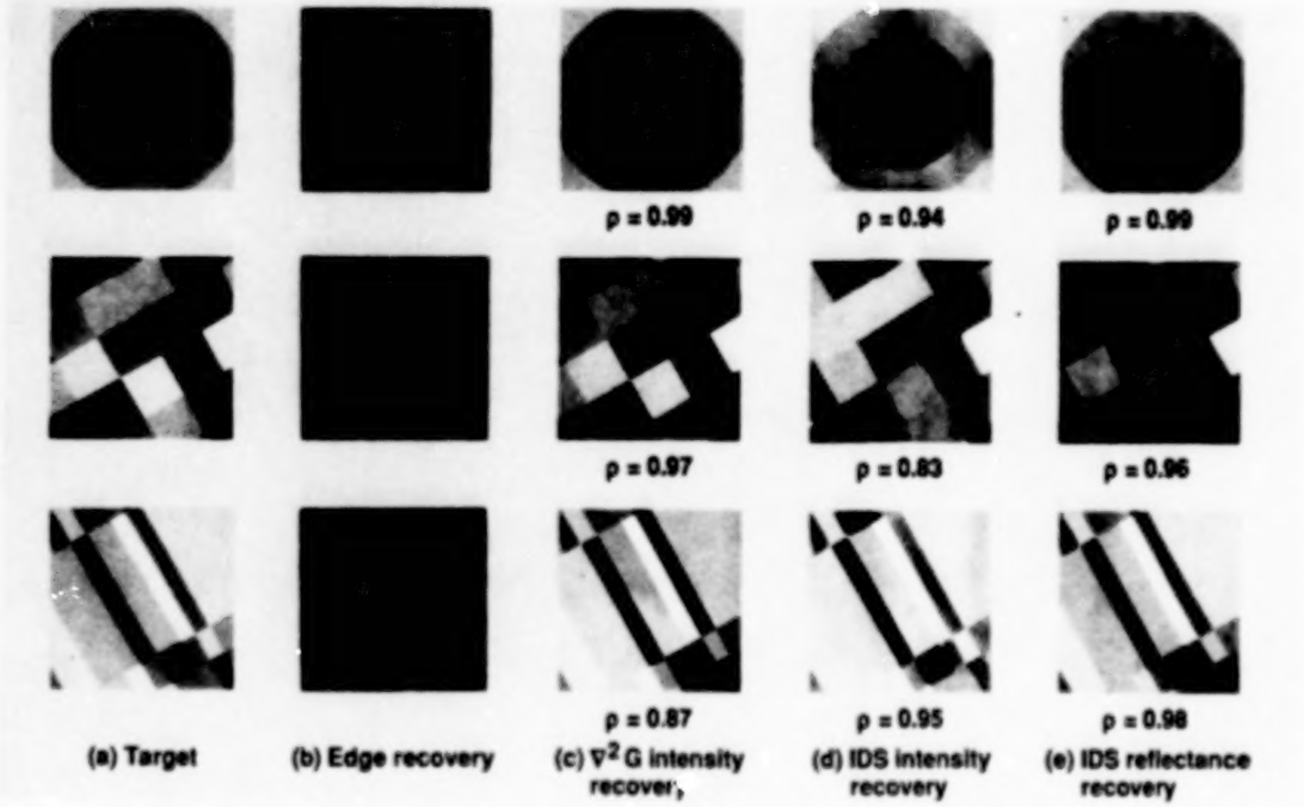
(a) Recovery of W

(b) W = 5/I

(c) W = max $\left[\frac{255-I}{I}, 1\right]$

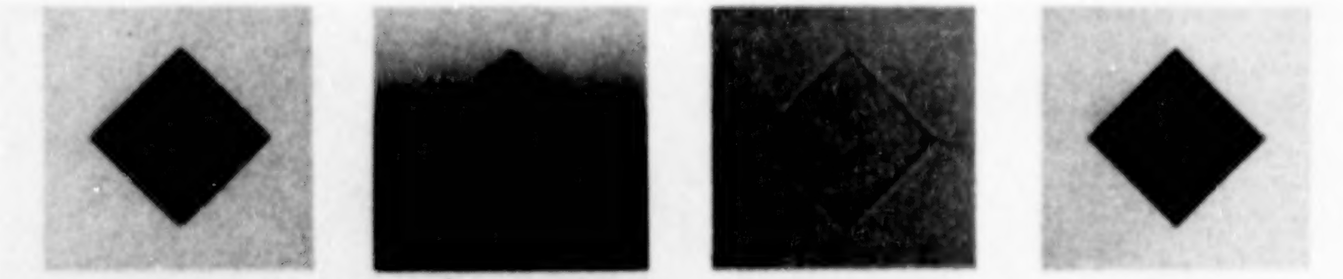
(d) W = 0.5

Figure 8: Accuracy of the IDS recovery process.



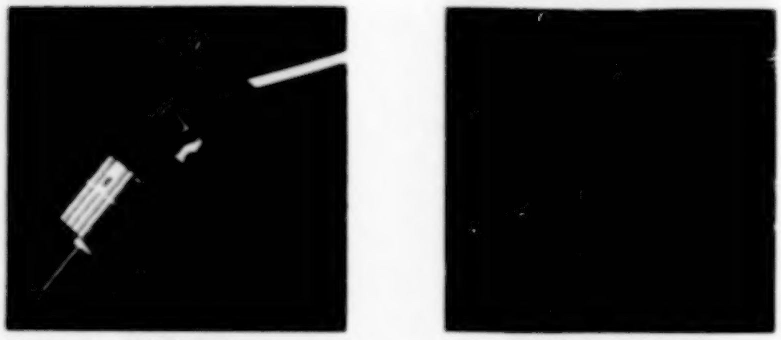
(a) Target (b) Edge recovery (c) $\nabla^2 G$ intensity recovery (d) IDS intensity recovery (e) IDS reflectance recovery

Figure 9: The recovery process for computer-generated targets.



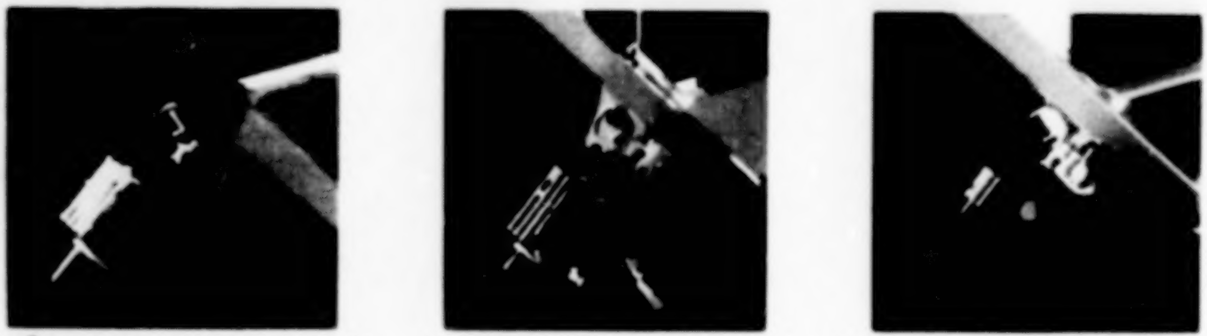
(a) Original target with constant illumination (b) Original target with spatially varying illumination (c) IDS response to target (b) (d) Reflectance recovery

Figure 10: Recovery from spatially varying illumination.



(a) Target

(b) Edge recovery



(c) $\nabla^2 G$ intensity recovery. $\rho = 0.84$

(d) IDS intensity recovery. $\rho = 0.45$

(e) IDS reflectance recovery. $\rho = 0.42$

Figure 11: Feature extraction for an experimental image that simulates imaging conditions in space.

APPLICATIONS OF THE IDS MODEL†

Eleanor Kurrasch
Odetics, Inc.
1515 S. Manchester Ave.
Anaheim, CA

INTRODUCTION

The theory of the IDS model was discussed earlier by Dr. Tom Cornsweet (ref. 1). This Intensity-Dependent Spread algorithm originated with Dr. Tom Cornsweet and with Dr. John Yellott. They introduced this concept of a spatially variant image processing technique that is like the human visual system (ref. 2). Odetics furthered this research by applying the IDS model to some very interesting applications. This model has some unique characteristics, as described elsewhere, and is a nonlinear spatially adaptive bandpass filter that is locally adaptive and robust to signal noise. Odetics' research was to evaluate the IDS model for processing at video rates. The approach was to develop a prototype of a VLSI video rate version and apply IDS to machine vision. IDS could provide very significant bandwidth reduction in the transmission of information from a remote sensor. The IDS concept seems to explain the apparently complicated and unrelated visual phenomena exhibited by the human retina. Figure 1 shows the comparison of a typical image processing system with the human retina.

In Figure 2 some of the typical problems with digitized images are listed. The IDS model offers a solution to these types of problems. Odetics developed the research which consisted of a detailed performance analysis of the IDS model and a feasibility study of a hardware design and implementation, one in analog and one in digital. The analog approach is similar to a neural network with analog processors at each pixel.

The photograph in Figure 3 illustrates the output of the prototype 9x9 analog system for a simple T-shaped image. This analog implementation of the IDS model has 81 pixel fiber optic inputs, 81 analog processor circuits, and an LED display. The analog circuits limited the dynamic range of the output and worked well enough to prove the feasibility of this concept. The next approach was a digital implementation of the IDS model that would execute at video rates. Odetics developed a detailed design during the Phase I research for LaRC.

---

†The following work was supported by NASA Contracts NAS1-18468 and NAS13-339.

The results of the performance analysis conducted on the IDS model during the research demonstrated significant advantages for both high bandwidth and low intensity scenes over small kernel operators such as Sobel and spatially invariant large kernel operations such as Marr-Hildreth. The research verified the predicted performance of the model with both simulated and actual images and showed that IDS could provide the edge enhancement necessary for machine vision systems. IDS would also provide significant bandwidth reduction. The most exciting result was the development of digital design approach for a general purpose large kernel digital convolver which can implement spatially variant IDS spread functions as well as common spatially invariant ones. Our results showed the feasibility of implementing the convolver to perform IDS processing on a 512 x 512 pixel image at video rate. Results are contained in the Phase I NASA final report reference (ref. 3).

Figure 4 illustrates the fundamental concept of the 32 x 32 bit convolver where each output pixel is the result of the simultaneous processing of 32 x 32 or 1024 input spread functions. This operation is performed in about 100 nanoseconds per pixel. In Figure 5 we show the results of Phase I and the proposed results of Phase II and the deliverables. In Figure 6 we describe the proposed detailed characteristics of the prototype of the IDS VLSI image processor. It will operate at video rate with 512 x 512 8-bit video format and be capable of handling a 32 x 32 spread function. It has a 16-bit internal representation for intensity mapped pixels. An interim prototype was developed on 1 PC board which operates at 4 seconds per frame. The video rate version is under development. To increase the dynamic range an advanced version is being designed using 12 bit video input data, operating on a 64 x 64 or even 128 x 128 spread functions. The objective of this advanced version is a video rate operation in VLSI and space qualification. Figure 7 shows the interim near-real-time version which operates at 4 sec/frame. Figure 8 illustrates two video input images taken in the laboratory under very low light. The top left is a small tank and the top right is the PC board layout of this interim IDS processor. In the lower left we see the result of the IDS processing and the edge enhancement of the tank and the lower right shows the IDS performance on the PC layout board design.

Figure 9 illustrates the Vision Workstation which we are delivering to LaRC. It consists of a SUN 3/260C computer and a Datacube pipeline video processor. The IDS processor will consist of a board set that is designed for the Datacube architecture.

Odetics is under contract to NASA at the John Stennis Space Center which exploits a very exciting characteristic of the IDS model which may be used in what is termed color constancy. Color vision is characteristic of a small number of living species. These creatures have used color to adapt and survive in the world around them. When a person is asked to identify objects in a scene or in a color photograph of a scene, he or she may use any of a large number of possible cues. One of the most salient and useful cues is the color of an object. In this paper we refer to the term, "color" as spectral reflectance. The identification of an object requires prior recognition or determination that a particular region of a scene is an object. This is defined as object segmentation. Many objects have a relatively uniform color that is different from the background, and in those cases, color can provide important input for segmentation. All contiguous pixels, within some given area, can then be grouped together and labeled as a single object. Object segmentation is performed effortlessly by the human visual system, but creating computer vision that takes an image as input and performs object identification on the basis of color runs into difficulties. Automatic identification is difficult enough when the conditions of illumination on the scene are precisely known, but as an example in images acquired by satellites, several unknown aspects of the light illuminating the scene can seriously interfere with the use of color in object identification. The color of an image depends not only on the physical characteristics of the object but also on the wavelength composition of the incident illumination. The color of the image of an object, particularly when there is a large distance between the object and the imaging system, is also strongly affected by the spectral transmission and the scattering characteristics of the air that lies between the object and image. Those characteristics in turn depend on the amounts of moisture, dust, smoke, etc., that are suspended in the air. For these reasons, the colors of the images of objects do not directly indicate the colors of the objects themselves, and the use of image colors in the segmentation and identification of objects is severely compromised. The fact that human vision is much less affected by changes in the color of the illumination than television or photographic image recording systems has been known in the human vision literature for a long time and is called color constancy. This term means that the color of an object is relatively constant in spite of changes in the color of the illuminant.

IDS processing provides the extraction of edges and of reflectance changes across edges, independent of variations in scene illumination. Suppose then that IDS processing were carried out at video rate in the camera and that only the information actually transmitted from the camera to a receiver were the locations of each edge pixel and the ratio of reflectance across the edge. A useful image of the objects in the scene could then be reconstructed at the receiver. For most scenes, that process would provide very significant band-width reduction while performing useful feature extraction (relative reflectance, independent of illumination) at the same time. IDS yields edge responses whose amplitudes are independent of scene illumination and depend only upon the ratio of the reflectances on the two sides of the edge (ref. 4).

The color constancy test process is shown in Figure 10. In the Phase I contract we applied the IDS concept to multispectral scenes in order to demonstrate the capability to determine the correct color of objects and patterns in the scene independent of the color of the illumination on the scene. In the Phase II contract, we are using the IDS output to reconstruct the reflectance image. We are also developing a color constancy testbed as shown in Figure 11 which will use the IDS video rate processors with a SUN workstation and Datacube processor. Here we will be able to test and evaluate outdoor images using the IDS concept. Figure 12 shows four mondrian images. In the upper left is the original input image. In the upper right is an earlier attempt at developing the reconstruction algorithm which shows much interference at the corners. In the lower left is a later attempt that assumes that there is a relationship such that the peak-to-trough amplitude and step functions are the same at both ends and that the step size is linear. The lower right has a further modification of this algorithm. The objective Odetics has for this Phase II research is to design and build a multispectral camera testbed to generate color constant images in near-real-time and perform color reflectance image reconstruction. A product which may be developed by Odetics in the future is a CCD camera containing an advanced IDS (VLSI) processor that will exhibit the features we've discussed here and be used for object recognition. Figure 13 shows an artist's sketch of the camera.

REFERENCES

1. Tom N. Cornsweet, Prentice Award Lecture: "A Simple Retinal Mechanism That Has Complex and Profound Effects on Perception". American Academy of Optometry, School of Social Sciences, University of California, Irvine, CA (1985).

2. Tom N. Cornsweet and John I. Yellott, Jr., "Intensity-Dependent Spatial Summation", Cognitive Science Group, University of California, Irvine, CA.

3. Adaptive Focal Plane Processor for Image Enhancement, Final Report, NASA Contract NAS1-18204, September 12, 1986.

4. Advanced Pattern Recognition Techniques in Image Analysis, Final Report, NASA Contract NAS13-302, October 8, 1987.

SPECIFICATIONS	TYPICAL IC	RETINA
• CIRCUIT LAYOUT	• 2-DIMENSIONAL	• 3-DIMENSIONAL
• IC LINE WIDTH	• 1-3 MICRONS	• 0.1 - 1.0 MICRON
• NUMBER OF GATES	• APPROX. 1,000,000	• APPROX. 25,000,000,000
• RESOLUTION (PIXELS)	• 2048 X 2048	• 10,000 X 10,000
• POWER CONSUMPTION	• 200 - 300 WATTS	• 0.001 WATTS
• SYSTEM VOLUME	• APPROX. 10,000 CU. IN.	• APPROX. 0.0003 CU. IN.
• TOTAL WEIGHT	• 20,000 - 50,000 G	• < 1 G

Figure 1 Comparison of Typical Image Processing System with the Human Retina

- RANGE OF NATURAL ILLUMINATION IS ONE TO TEN BILLION
  - -- LOW ILLUMINATION - NIGHT VISION
  - -- HIGH ILLUMINATION - BRIGHT SUNLIGHT
- LOSS OF DYNAMIC RANGE
- LOSS OF SPECTRAL RESOLUTION
- SHADOWS AND VARYING ILLUMINATION ON THE SAME IMAGE
- NOISE
  - -- BACKGROUND NOISE
  - -- SENSOR NOISE
- LOSS OF HIGH SPATIAL FREQUENCY INFORMATION
  - -- OBJECTS (EDGES) NOT CLEAR

Figure 2 Typical Problems with Digitized Images

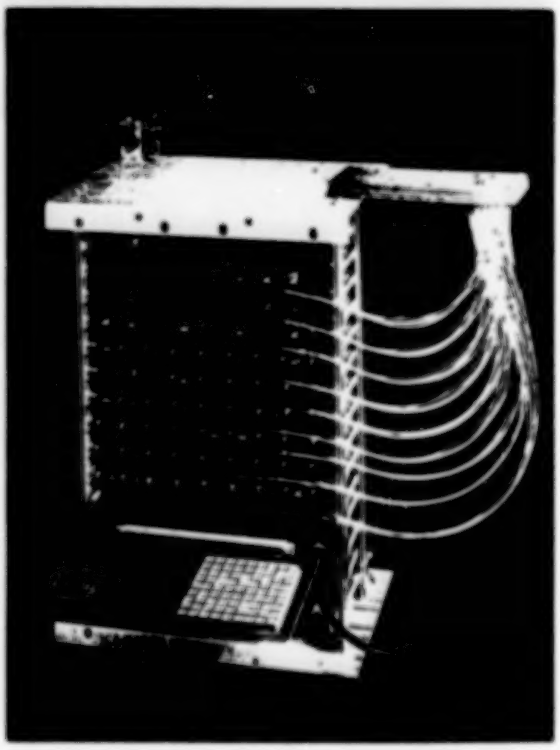
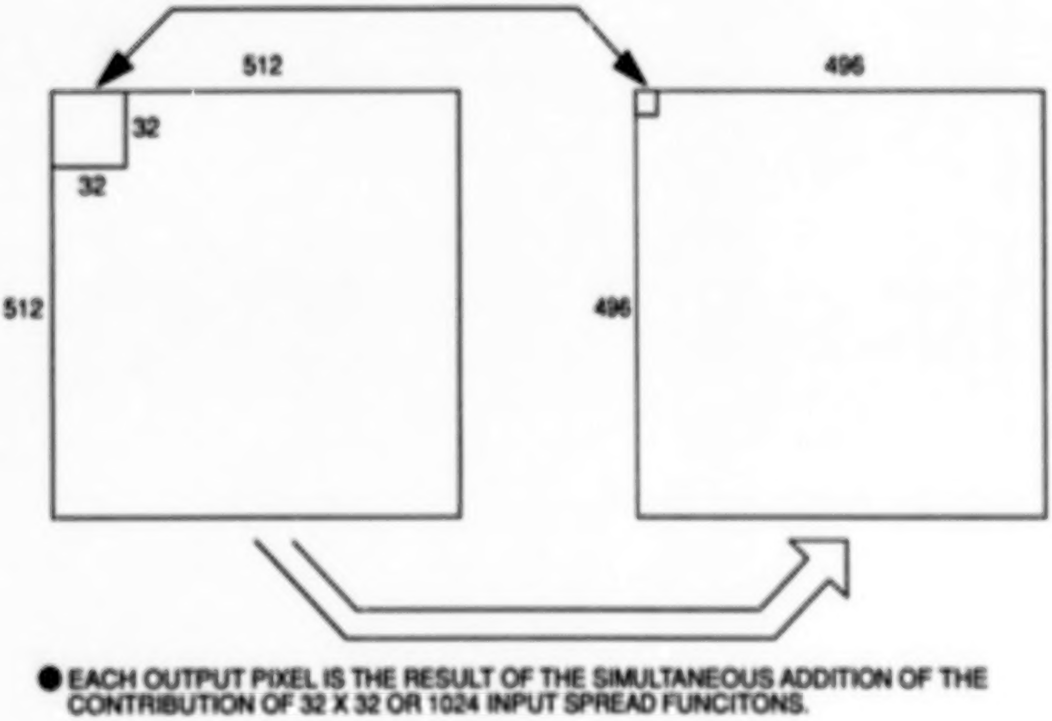


Figure 3 A 9x9 Analog Implementation of IDS



● EACH OUTPUT PIXEL IS THE RESULT OF THE SIMULTANEOUS ADDITION OF THE CONTRIBUTION OF 32 X 32 OR 1024 INPUT SPREAD FUNCITONS.

Figure 4 32x32 Convolver

DELIVERABLE

PROTOTYPE PROCESSOR WITH:

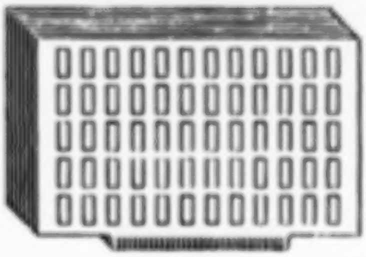
- Spatially invariant filtering (conventional)
- Spatially varying filtering (IDS or other)
- Video rate w/wo interlace
- Kernel from 1 x 1 to 32 x 32 pixels

POTENTIAL:

- VLSI in one 5" x 5" board with 8 chips

PHASE I
Demonstration
(Analog)

PHASE II
Deliverable
(Digital)
(8 Boards)



POTENTIAL
(VLSI)
(1 Board)

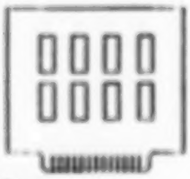


Figure 5 Phase I Phase II Deliverables

- **Real Time Processor of 512 x 512 8-Bit Video Format**
- **32 x 32 Spread Function**
- **16-Bit Internal Representation for Intensity Mapped Pixels**
- **Giga-pix / Sec Processing Rate**
  ($512 \times 512 \times 30$ f/s $\times (32 \times 32) => 8.053 \times 10^9$)
- **Near Real Time Prototype (1 PC Board) (4 sec/frame)**
- **Real Time Prototype (8-16 PC Boards)**
- **Proposed VLSI Version (8 Chips)**

Figure 6 IDS VLSI Image Processor (Convolver)

Figure 7 was unavailable at time of publication. Upon written request, interested parties can procure a copy from Odetics at a later date.

Figure 7 Interim Near-Real-Time Version of FPPJr.

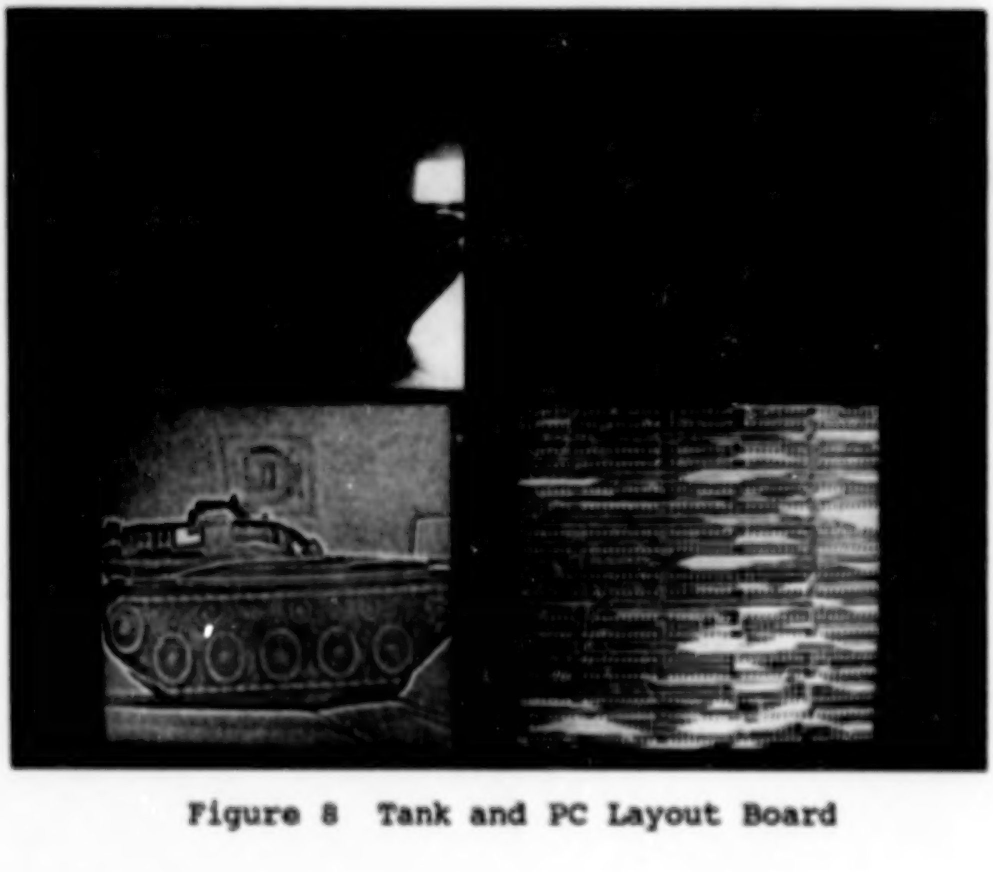


Figure 8 Tank and PC Layout Board

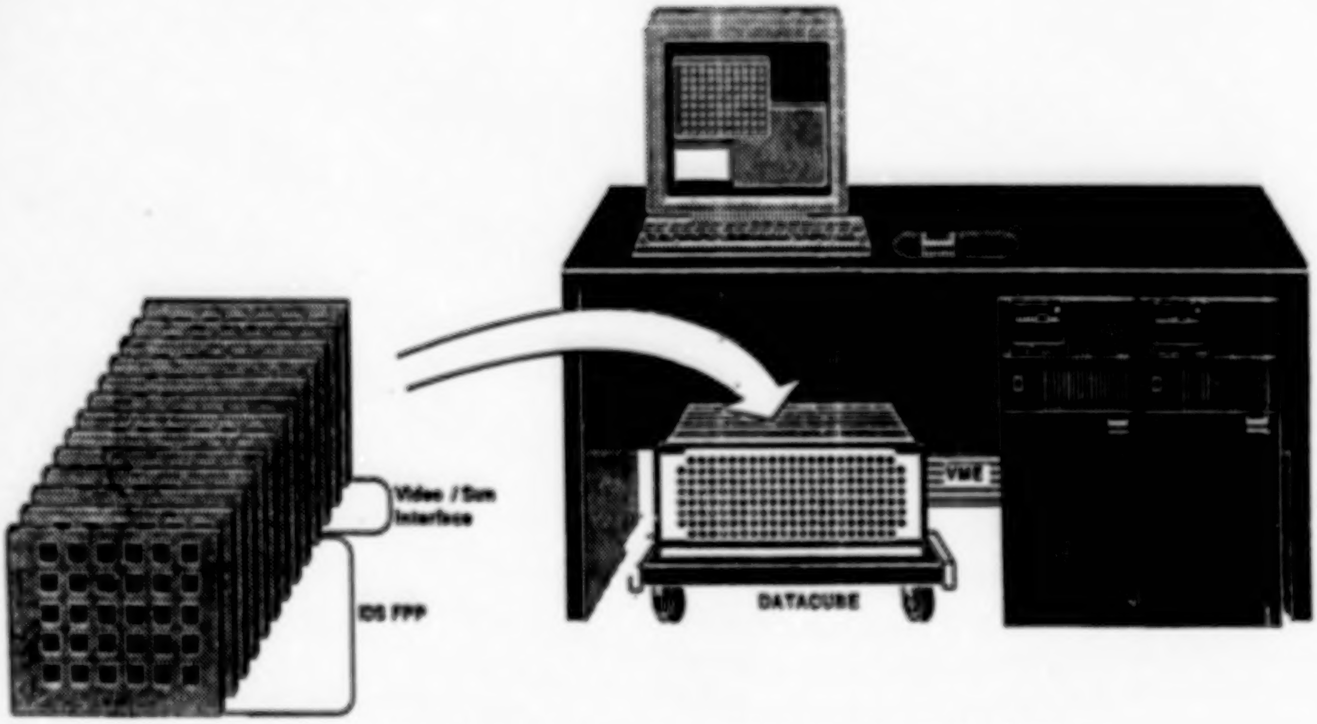

Video / Sun
Interface
IDS FPP
VME
DATACUBE

Figure 9 Real-Time IDS Vision Workstation

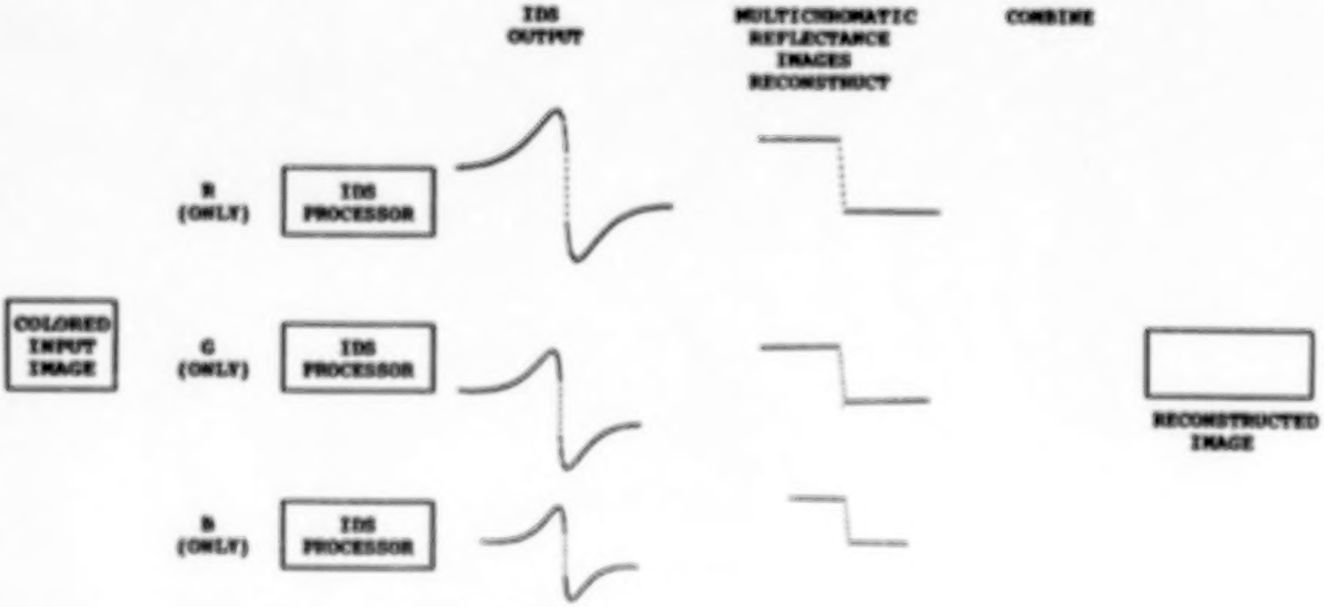

IDS
OUTPUT
MULTICHROMATIC
REFLECTANCE
IMAGES
RECONSTRUCT
COMBINE
R
(ONLY)
IDS
PROCESSOR
COLORED
INPUT
IMAGE
G
(ONLY)
IDS
PROCESSOR
RECONSTRUCTED
IMAGE
B
(ONLY)
IDS
PROCESSOR

Figure 10 Color Constancy Test Process

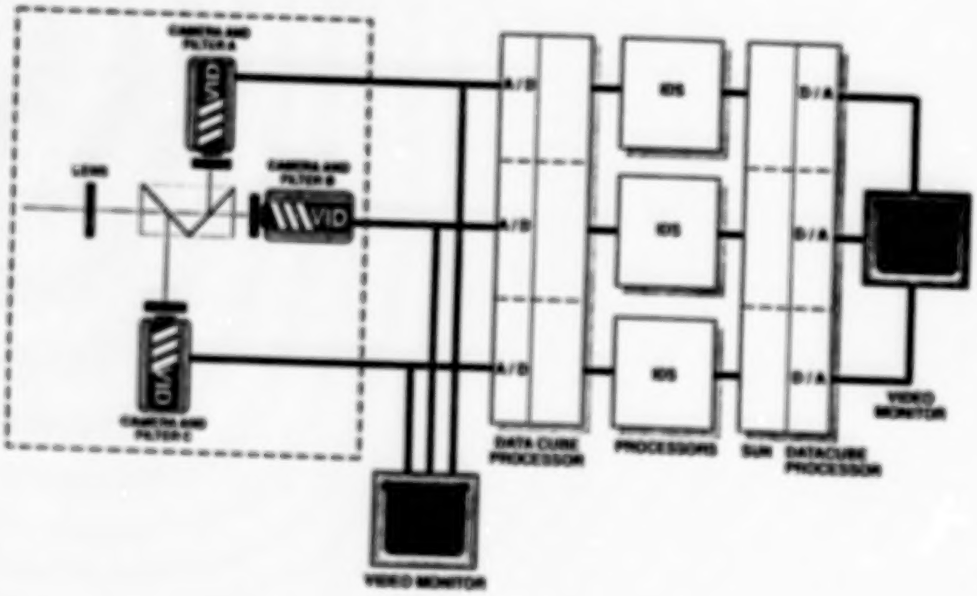

CAMERA AND FILTER A
VID
LENS
CAMERA AND FILTER B
VID
VID
CAMERA AND FILTER C
A/D
A/D
A/D
IDS
IDS
IDS
D/A
D/A
D/A
VIDEO MONITOR
DATA CUBE PROCESSOR
PROCESSORS
SUM
DATACUBE PROCESSOR
VIDEO MONITOR

Figure 11 Color Constancy Test Bed

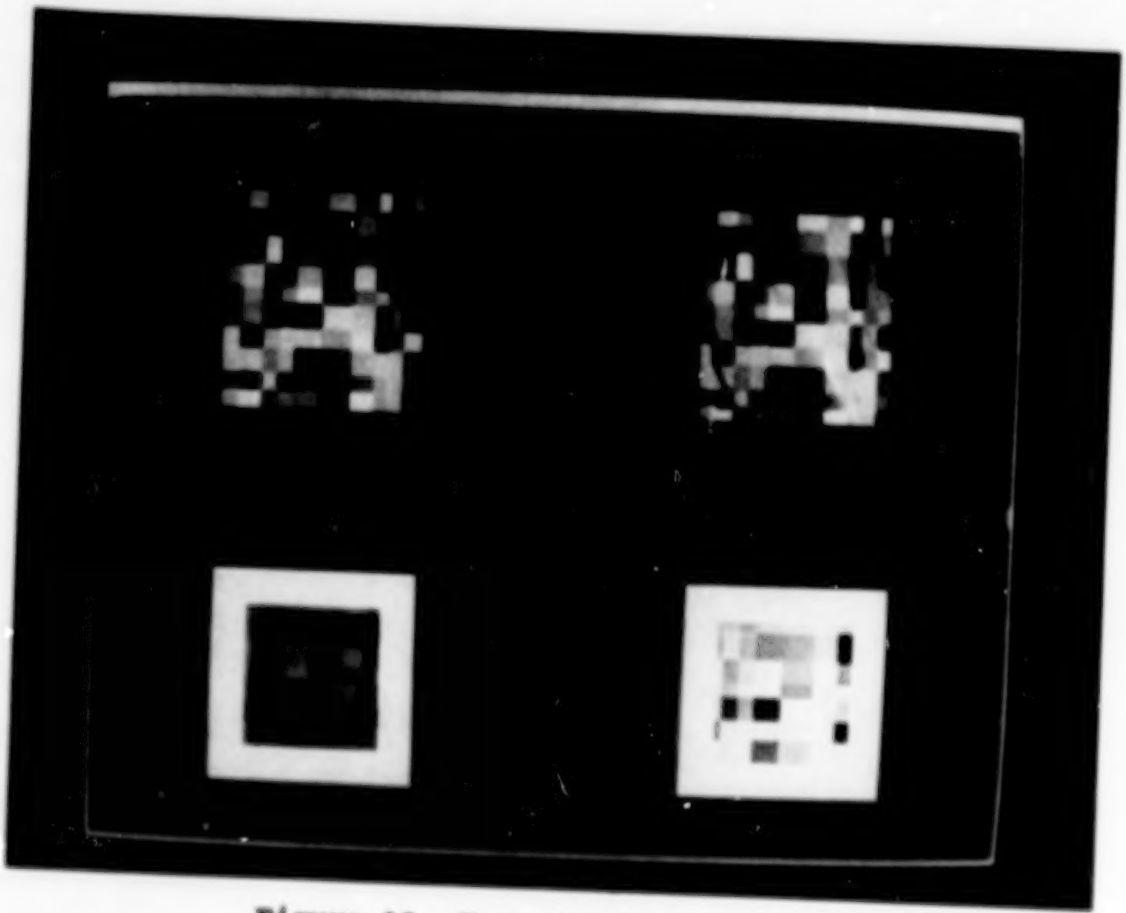


Figure 12 Four Mondrian Images

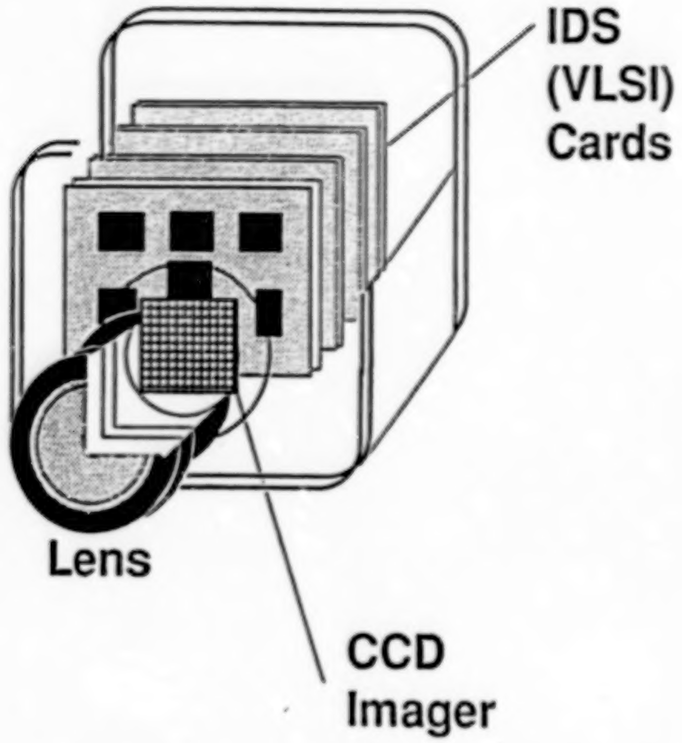

IDS
(VLSI)
Cards
Lens
CCD
Imager

Figure 13 IDS Camera (VLSI)

ISOLATING CONTOUR INFORMATION FROM ARBITRARY IMAGES

Daniel J. Jobson
NASA Langley Research Center
Hampton, Virginia

Abstract

Aspects of natural vision (physiological and perceptual) serve as a basis for attempting the development of a general processing scheme for contour extraction. Contour information is assumed to be central to visual recognition skills. While the scheme must be regarded as highly preliminary, initial results do compare favorably with the visual perception of structure. The scheme pays special attention to the construction of a smallest scale circular difference-of-Gaussian (DOG) convolution, calibration of multiscale edge detection thresholds with the visual perception of grayscale boundaries, and contour/texture discrimination methods derived from fundamental assumptions of connectivity and the characteristics of printed text. Contour information is required to fall between a minimum connectivity limit and maximum regional spatial density limit at each scale. Results support the idea that contour information, in images possessing good image quality, is contained largely if not wholly in the highest two spatial frequency channels (centered at about 10 cyc/deg and 30 cyc/deg). Further, lower spatial frequency channels appear to play a major role only in contour extraction from images with serious global image defects.

Introduction

The goal of sophisticated machine vision capabilities requires that attention be paid to the semantic content of images together with intrinsic image characteristics such as contrast, noise, blur, and sampling. The visual task of recognition and interpretation is of paramount interest. Here the first step toward a more general approach to machine vision recognition is defined as a set of methods for transforming arbitrary images into contour or schematic line drawing information. This set of methods was fashioned from natural vision concepts (physiological and perceptual) coupled with image acquisition processes. Emphasis is placed on the smallest scales of the image. An abbreviated pyramid of circular difference-of-Gaussian (DOG) convolution operators forms the first stage of image processing. In particular the nontrivial problem of constructing a smallest scale DOG operator from discrete image samples is of special interest. Edge detection and contour extraction processing is then performed for each scale of operator. Multiscale contour information is then merged in an hierarchical manner with priority being given to the information from the smallest scale. Larger scale information is only added to spaces not already occupied by smaller scale information.

The set of methods can be viewed as a progression starting with the image and proceeding through "seeing" to significance. The processing scheme, which must be regarded as preliminary, is described together with the ideas behind the scheme. Results are given for a series of image processing experiments designed to provide a partial demonstration of the overall consistency of the scheme with the visual perception of structure in images. While conciseness of either the processing or the resulting information was not a major goal, it is noted that processing is reasonably simple and the contour information is intrinsically concise (and can be made more compact by the addition of coding schemes).

Construction and Resiliency of Smallest Scale Operator

Previous work by Huck et al. (Ref. 1) demonstrated that a well-behaved smallest scale DOG operator could be constructed for one case of a specific amount of image blur and a particular spacing of the square grid sampling lattice. Subsequently this work was extended to show that this well-behaved operator will result from the same set of weighting coefficients even for significant changes in blur or sampling lattice spacing (Ref. 2, Figs. 1 and 2). The two-dimensional convolution of these weighting coefficients with the image samples is then equivalent to applying a small DOG operator to the original scene radiance distributions. Less attention was paid to the construction of larger operators (in this case about 3 times and 6 times larger) and the larger functions used were merely formed from discrete values of the desired size of DOG function. The spatial spread of image edges after convolution was checked as a rough verification that the desired scale operator was achieved.

Multiscale Two-Dimensional Edge Detection and Representation

A fully two-dimensional edge detection method was found to be necessary (Ref. 2). Likewise an edge representation space magnified by a factor of two over the original image space was also a requirement (Figs. 3, 4, and 5). Edge detection is based on zero-crossings (Ref. 3); however, the determination of thresholds for "zero" for edge detection could not be determined from fundamental considerations. In particular, noise-limited edge detection produced edge representations with a wealth of textural detail which seemed inconsistent with the subsequent goal of contour extraction. Therefore, the performance of visual perception was examined for guidance in determining multiscale contrast sensitivities. To this end, the perception of grayscales edges and bar patterns was examined.

The Disparity Between Grayscale and Bar Pattern Perception at Scales Larger than the Visual Acuity Limit

The perception of both grayscale and bar patterns (Fig. 6) seemed suitable to the calibration of contrast sensitivity versus image scale. One aspect of grayscale edge perception is noteworthy. For equal step intervals in the grayscale and decreasing angular size, a point is reached where almost all edges vanish at once. The exceptions are the lowest and highest steps which vanish at slightly larger angular sizes. Therefore for a particular angular size, edge detection in grayscales seems to be an almost linear process with constant threshold value.

While a consistent result for grayscale edges and bars was expected, actual results were quite different. A striking disparity occurred between the perception of grayscale edge and bar patterns at 3x and 6x the visual acuity limit (Fig. 7). This led to the use of the grayscale sensitivities for edge detection and the formation of a hypothesis that contour information exists as a higher contrast subset of information within the full range of visual phenomena (Fig. 8). It should be noted in these spatial frequency diagrams that higher contrast at a given scale is a necessary but not sufficient requirement for visual phenomena to be contour information. That is, some higher contrast phenomena may still prove to be textural or otherwise not relate to overall contour description of a scene. Contrast sensitivity versus scale must now be related to edge detection zero-crossing thresholds by considering noise, blur, edge contrast, and most importantly sampling effects.

Edge Detection Threshold-Calibration to Contrast Perception Considering Sampling, Noise and Blur

Sampled edge convolution signals exhibit considerable chatter compared to the characteristic analog signal (Fig. 3). Therefore, capture of extended edges in a test image at each scales' contrast sensitivity was calibrated for intrinsic sampling errors coupled with reasonable values of noise and blur. The existence of reasonably low noise (S/N > 50) and modest blurring (Gaussian $\sigma = 0.6$ of the sampling lattice spacing) was checked. This calibration (Table 1) was performed in two stages. A set of convolution samples on extended edges at threshold contrast was used to make an estimate of edge detection threshold. Since this relatively small sample might not be highly accurate, some image processing experiments were performed. Edge detection thresholds were adjusted until the bulk of extended straight edges at minimum contrast were detected.

Contour Extraction Methods (Semantic Reference Points)

To this point only the "seeing" of edges has been considered. The isolation of contour information now necessitates a step beyond this into the semantic content of contour information. The question is namely - can we find a fundamental characteristic of contour information on which further processing can now be based? Contour information subjectively seems to always steer between "too little" and "too much", so we can look for ways to establish quantitative criteria for these subjective limits. Connectivity was investigated as a "too little" criterion while regional spatial density of events was used for a "too much" criterion. The following two approaches to minimum connectivity were investigated: 1) the minimum feature of printed text - the period, or more fundamentally 2) connectivity across the space occupied by the original DOG operator. The latter proved to be the most perceptually consistent. Printed text characteristics were also investigated to establish a region size and a maximum number of spatial events allowed. This hypothesis arose from the idea that printed text is engineered by man for possibly maximum information throughput. The resulting quantitative criteria are summarized in Table 2.

Results, Discussion, and Conclusions

As a partial demonstration of generality, the computational scheme is applied identically to diverse images. The original image is shown at the correct size to place each image pixel at about the visual acuity limit for a normal reading distance (Fig. 9). However, perceptual comparisons with these reproduced images are not particularly accurate because the contrast rendition of the original image cannot be maintained in publication. Only results for the 1x and 3x combined scales are shown since this appears to be sufficient for good quality images. Addition of 6x scale information appears to be unnecessary in this case and comes into play for images with global defects (weak contrast, severe blur or noise). The handling of defective images is the subject of an on-going investigation and seems to require a graceful shift to a pair of larger scales as one or more smaller scales produce insufficient information in some global sense.

In an overall sense, these results support the idea that a general scheme for contour extraction is possible and can be based mostly on a pairwise selection of two scales of edge detection and representation. These two scales should be the smallest two for most normal imagery and shift to pairs of successively larger scales only when globally defective images occur.

References

1. Huck, F. O.; Fales, C. L.; Halyo, N.; Samms, R. W.; and Stacy, K.: Image Gathering and Processing: Information and Fidelity. J. Opt. Soc. America, Vol. 2, No. 10, October 1985, pp. 1644-1666.

2. Jobson, D. J.: Spatial Vision Processes: From the Optical Image to the Symbolic Structures of Contour Information, NASA TP2838, November 1988.

3. Hildreth, E. C.: The Detection of Intensity Changes by Computer and Biological Vision Systems. Comput. Vis., Graph., and Image Process., Vol. 22, No. 1, April 1983, pp. 1-27.

TABLE 1. - EDGE DETECTION THRESHOLDS FOR THE THREE SMALLEST IMAGE SCALES (1X, 3X, AND 6X VISUAL ACUITY LIMIT)

SCALE	DESIRED CONTRAST THRESHOLD	ESTIMATED EDGE DETECTION THRESHOLD	ACTUAL EDGE DETECTION THRESHOLD
1X	50%	19%	16%
3X	15%	7.0%	5.5%
6X	5.5%	1.2%	1.2%

TABLE 2. - CONTOUR PROCESSING CRITERIA (IN MAGNIFIED EDGE REPRESENTATION SPACE)

SCALE	CONNECTIVITY	NUMBER OF EVENTS	REGION SIZE
1X	6	75	25 x 25
3X	18	260	50 x 50
6X	36	350	75 x 75

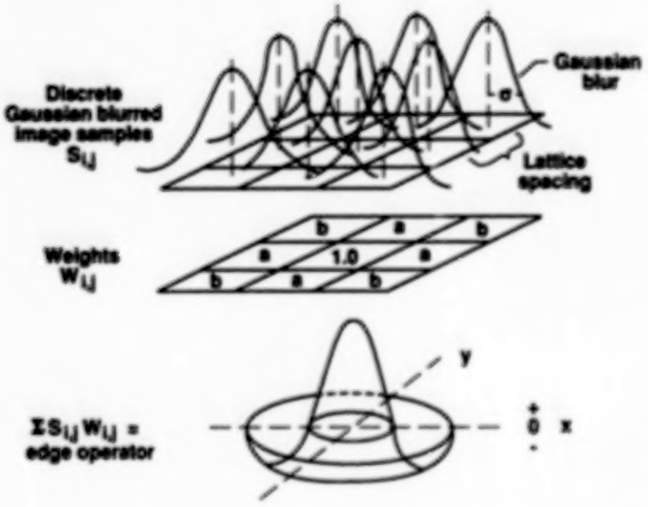

Discrete
Gaussian blurred
image samples
$S_{i,j}$
Gaussian
blur
σ
Lattice
spacing
Weights
$W_{i,j}$
b
a
b
a
1.0
a
b
a
b
y
$\Sigma S_{i,j} W_{i,j}$ =
edge operator
+
0
-
x

Figure 1. Construction of Smallest Scale Edge Operator for Square Lattice Image Space

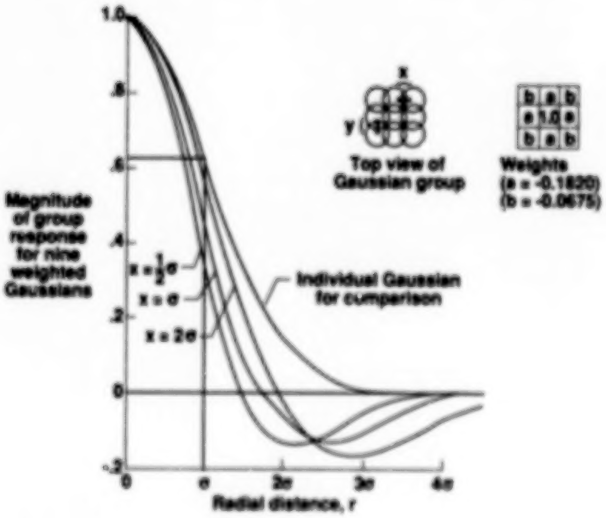


Figure 2. Resiliency of Operator to Variable Sampling Lattice Spacing (or Equivalently Variable Blur)

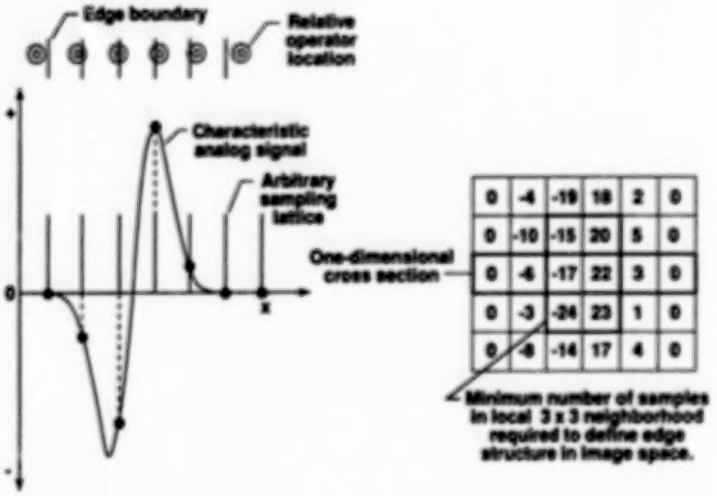

Edge boundary
Relative operator location
+
0
-
x
Characteristic analog signal
Arbitrary sampling lattice
One-dimensional cross section
0 -4 -19 18 2 0
0 -10 -15 20 5 0
0 -6 -17 22 3 0
0 -3 -24 23 1 0
0 -8 -14 17 4 0
Minimum number of samples in local 3 x 3 neighborhood required to define edge structure in image space.

Figure 3. Discrete Samples of Convolution of Edge with DOG Operator

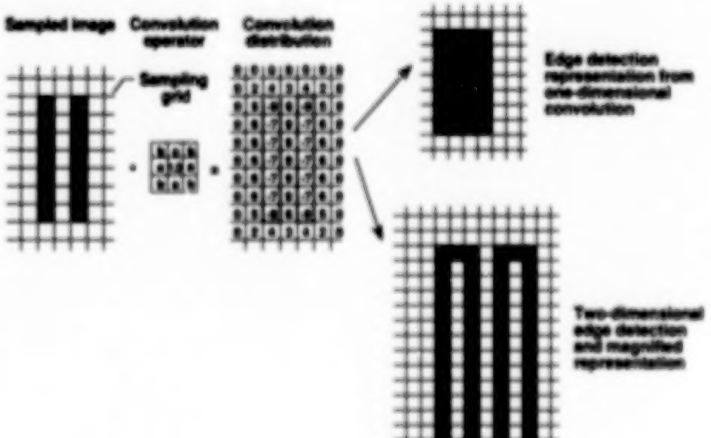

Sampled image
Convolution operator
Convolution distribution
Sampling grid
Edge detection representation from one-dimensional convolution
Two-dimensional edge detection and magnified representation

Figure 4. Illustration of the Requirement for a Magnified Edge Representation Space

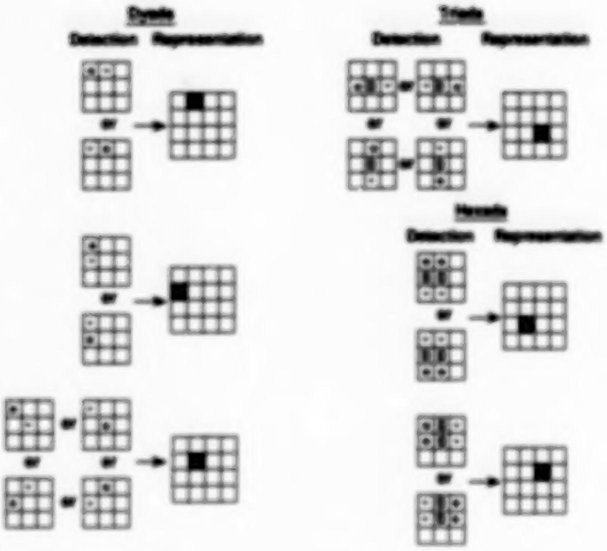

Dyads
Detection
Representation
Triads
Detection
Representation
Hexads
Detection
Representation
or

Figure 5. Zero-Crossing Comparisons Used in Edge Detection

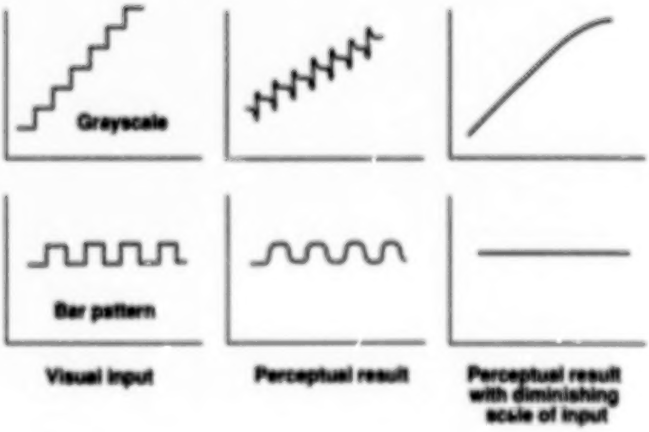

Grayscale
Bar pattern
Visual input
Perceptual result
Perceptual result with diminishing scale of input

Figure 6. Perceptual Determination of Contrast Sensitivity Versus Image Scale

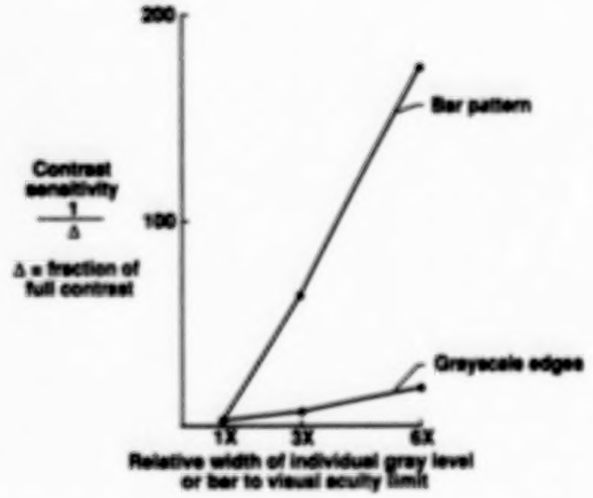


Figure 7. Disparity in Grayscale Edge and Bar Pattern Contrast Sensitivities for Scales Above Visual Acuity Limit

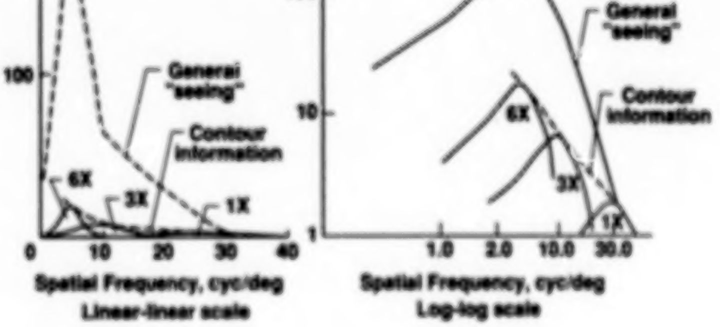


Figure 8. Hypothesis Regarding Contour Information

Figure 9. Image Processing Results

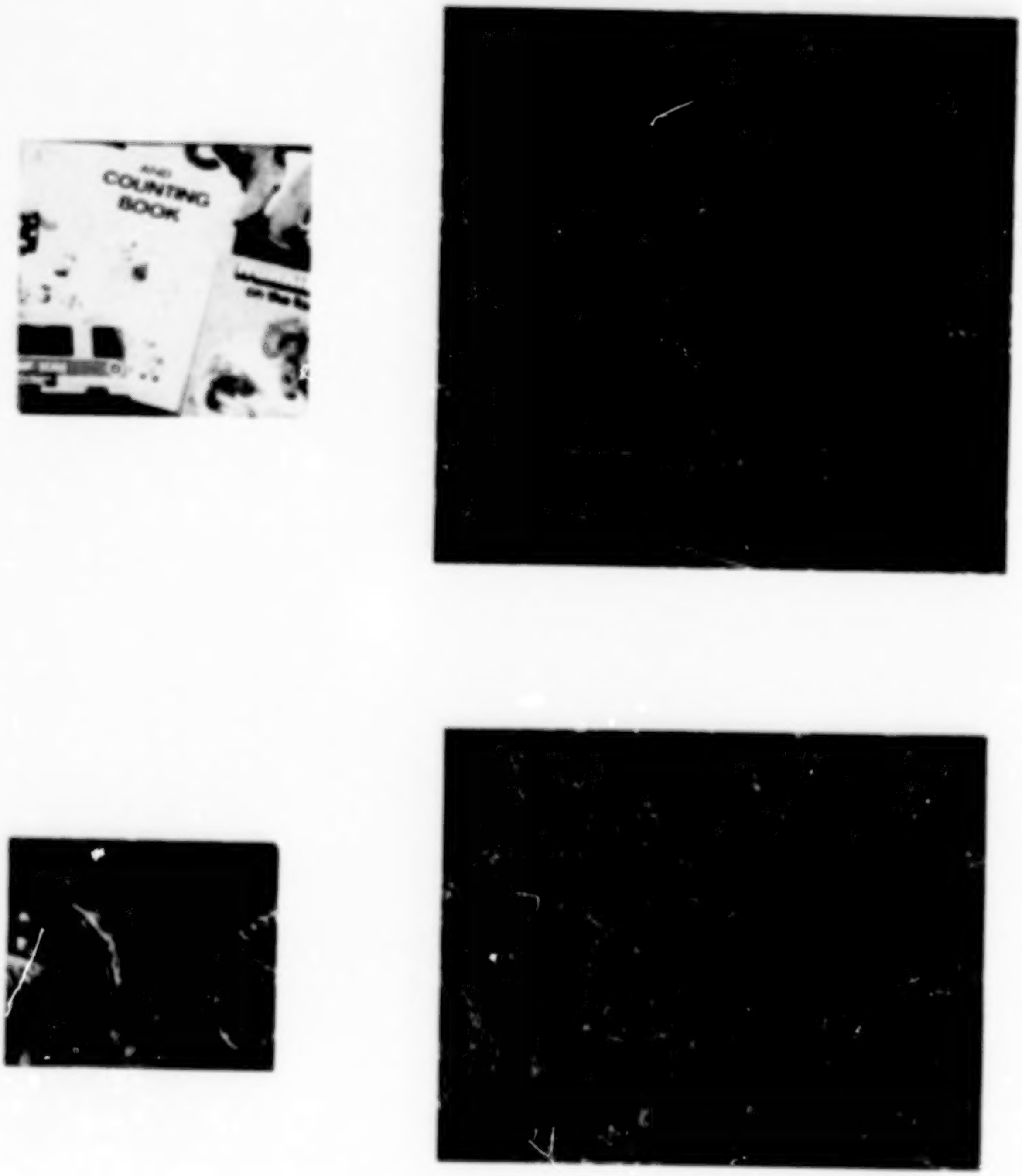

COUNTING
BOOK

Figure 9. Continued

loop control), which provide
simple display of target loo
rate, position error, warning
simplifying the maneuvering
required of the pilot. Ultim
of the powered free-flier to
may be required in order to
shocks to delicate user equip
experiments and the Space S

Figure 9. Continued

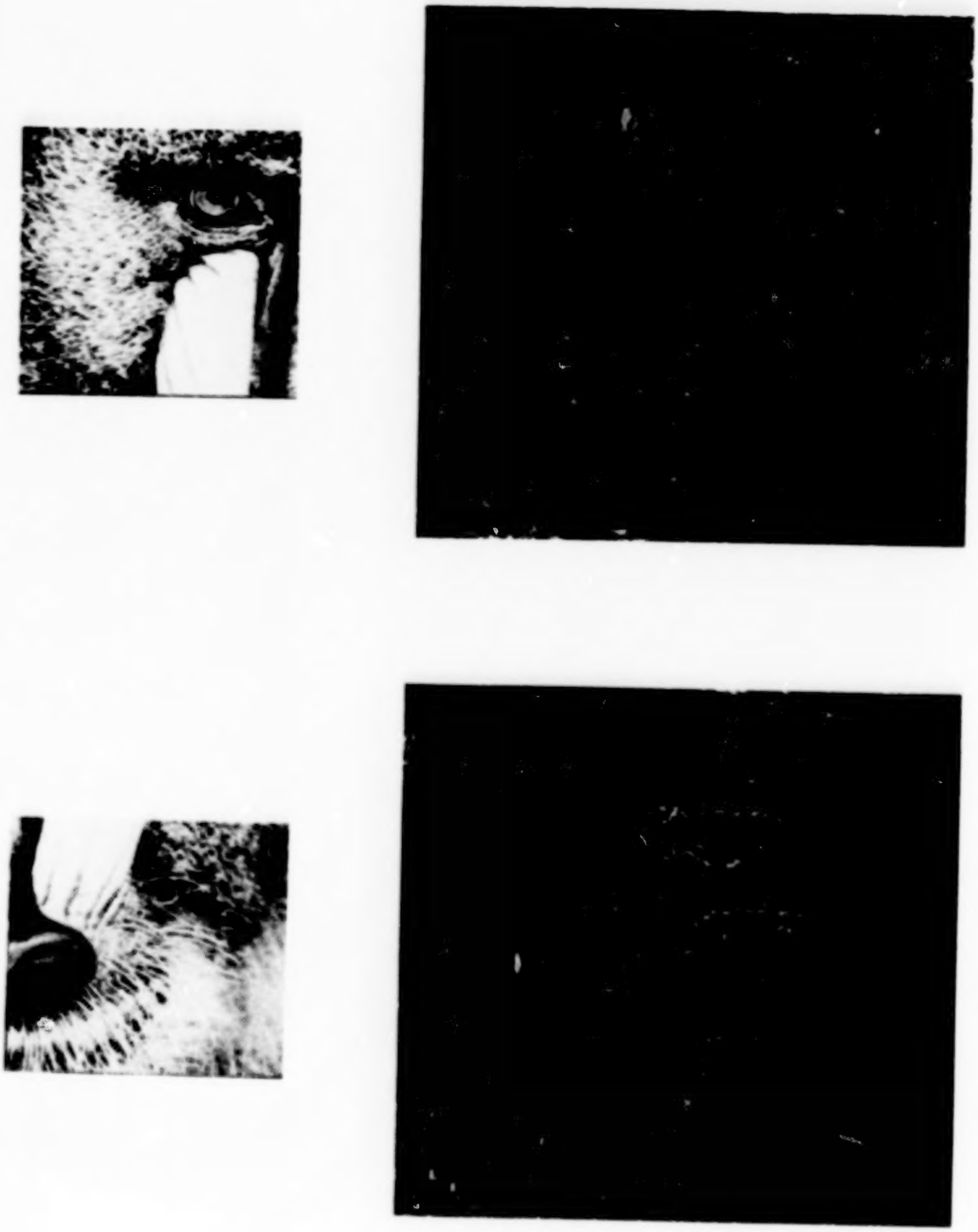


Figure 9. Continued

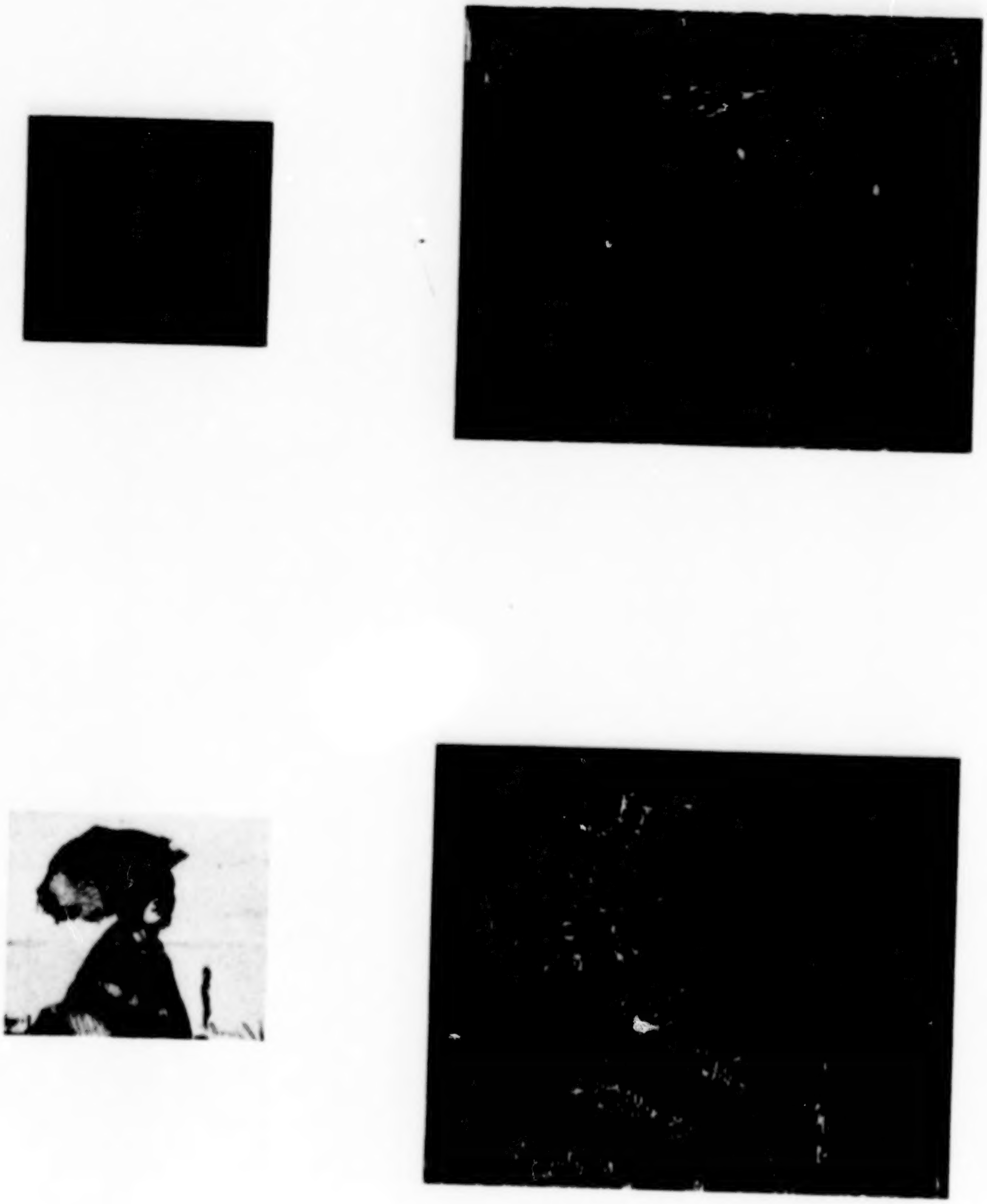


Figure 9. Concluded.

PARALLEL ASYNCHRONOUS SYSTEMS AND IMAGE PROCESSING ALGORITHMS

D. D. Coon
Microtronics Associates
4516 Henry Street
Pittsburgh, PA

and

A. G. U. Perera
Department of Physics
University of Pittsburgh
Pittsburgh, PA

Abstract

A new hardware approach to implementation of image processing algorithms is described. The approach is based on silicon devices which would permit an independent analog processing channel to be dedicated to every pixel. A laminar architecture consisting of a stack of planar arrays of the devices would form a two-dimensional array processor with a 2-D array of inputs located directly behind a focal plane detector array. A 2-D image data stream would propagate in neuronlike asynchronous pulse coded form through the laminar processor. Such systems would integrate image acquisition and image processing. Acquisition and processing would be performed concurrently as in natural vision systems. The research is aimed at implementation of algorithms, such as the intensity dependent summation algorithm and pyramid processing structures, which are motivated by the operation of natural vision systems. Implementation of natural vision algorithms would benefit from the use of neuronlike information coding and the laminar, 2-D parallel, vision system type architecture. Besides providing a neural network framework for implementation of natural vision algorithms, a 2-D parallel approach could eliminate the serial bottleneck of conventional processing systems. Conversion to serial format would occur only after raw intensity data has been substantially processed. An interesting challenge arises from the fact that the mathematical formulation of natural vision algorithms does not specify the means of implementation, so that hardware implementation poses intriguing questions involving vision science.

1 Introduction

Spontaneous generation of neuronlike action potential pulses in voltage or current driven silicon p^+-n-n^+ diodes at liquid helium temperatures has been studied extensively.[1,2,3,4,5,6] A simple circuit used to generate these pulses (Fig. 1) consists of a p^+-n-n^+ diode and a load resistor, capacitances and a current source.

In Fig. 2, we show how an optical sensor can be embedded in the circuit of Fig. 1. Such a circuit permits single stage coding of optical information into neuronlike spiketrains. The simplicity of the coding circuit would permit fully parallel, asynchronous processing of a two dimensional array of signals as would emerge from a 2-D array of photodetectors, i.e. a focal plane array. See Figs. 3 and 4.

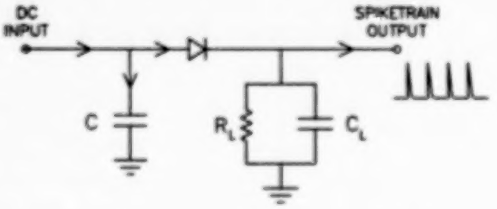


Figure 1: A circuit used to generate spontaneous neuronlike pulses. The only active circuit element is a p^+-n-n^+ diode.

Parallel asynchronous spiketrain signal processing would occur as in neural networks. The recent upsurge of interest in neural networks is an encouraging sign that the means of processing discussed here may be closely connected with significant new trends in signal processing and information processing.

By fully parallel processing, we mean one processing channel per pixel. This point is easily appreciated when one considers possible NASA image processing applications involving arrays of 1000 by 1000 pixels at 1 kilohertz frame rates. A fully parallel approach requires kilohertz processing in each channel while a fully serial approach would require processor speeds on the order of gigahertz. Processed output data may be much more condensed than raw input intensity data, so that conversion to a serial data stream after parallel processing is a very good strategy for many applications.

1.1 Hardware Implementation of Image Processing Algorithms

The above observations strongly suggest that our approach would be especially advantageous as a means of implementation of image processing schemes which are biologically motivated. An example of such an approach is given in the work of Marr and Hildreth[7] on edge detection and related general discussion of the computational viewpoint is given in Marr's influential book on vision.[8]

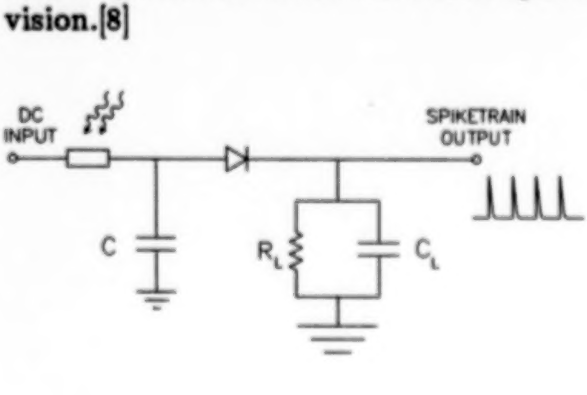


Figure 2: DC coupling of the current from a photodetector to a circuit which generates neuronlike output pulses. In the case of infrared illumination, a silicon impurity band detector is typically employed while in the case of near infrared, visible or ultraviolet illumination a reverse biased silicon p-i-n photodiode is employed.

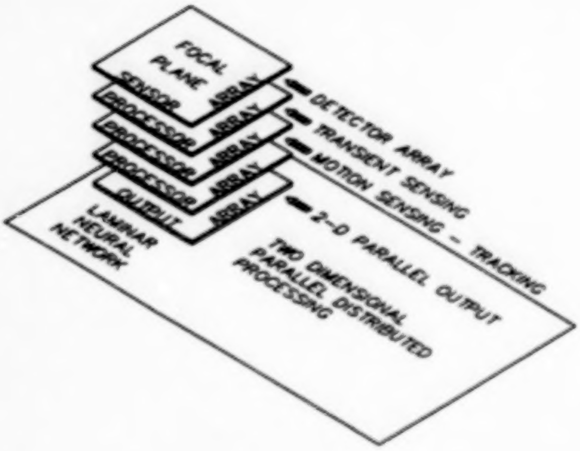

FOCAL PLANE ARRAY
SENSOR ARRAY
PROCESSOR ARRAY
PROCESSOR ARRAY
PROCESSOR ARRAY
OUTPUT ARRAY
LAMINAR NEURAL NETWORK
DETECTOR ARRAY
TRANSIENT SENSING
MOTION SENSING – TRACKING
2-D PARALLEL OUTPUT
TWO DIMENSIONAL PARALLEL DISTRIBUTED PROCESSING

Figure 3: Illustrative laminar architecture showing stacked wafers in three dimensions.

Biological image processing strategies and algorithms are of interest to NASA because of their potential for practical application. Of special interest is an algorithm of Cornsweet.[9,10] This Intensity Dependent Spatial Summation (IDS) algorithm is correlated with a number of quantitative, empirical aspects of vision.[11] The spatial scale of the associated point spread function is intensity dependent. Lower intensities are associated with a broader range of spatial integration. There is an interesting similarity between intensity dependent spatial integration and spiketrain coding of intensity in which the integration time is longer for lower intensities. Extended spatial integration and extended temporal integration are both strategies tradeoffs appropriate to coping with low intensity signal-to-noise problems.[12]

The Cornsweet algorithm is of interest in connection with edge detection, the identification of contours of objects and the specification of an image in terms of reflectance ratios. The temporal analog of a reflectance discontinuity at an edge is a step function intensity transient. One vision-system-like mode of transient sensing has already been demonstrated in our approach.[13] Implementation of the Cornsweet algorithm is a more subtle and interesting problem than transient sensing, although some insights may emerge from the similarity between spatial and temporal integration.

A parallel asynchronous hardware implementation of the Cornsweet algorithm would represent an interesting application of our approach. The ultimate and most challenging application would be real time, high frame rate, high resolution image processing. Hardware implementa-

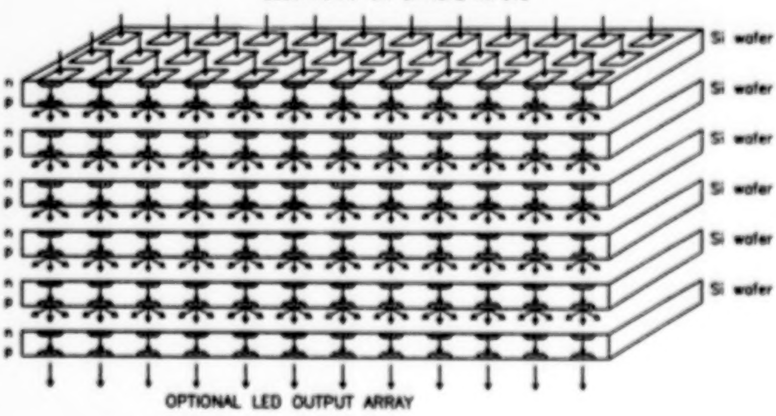

ELECTRONIC OR OPTICAL INPUTS
Si wafer
Si wafer
Si wafer
Si wafer
Si wafer
Si wafer
n
p
OPTIONAL LED OUTPUT ARRAY

Figure 4: Schematic illustration (cross-sectional side view) of the signal flow pattern through a 2-D parallel asynchronous processor consisting of stacked silicon wafers. Parallel asynchronous fire-through is a key to propagation of pulsed signals through chips. Injection pulses are associated with current flow between the n- and p-layers.

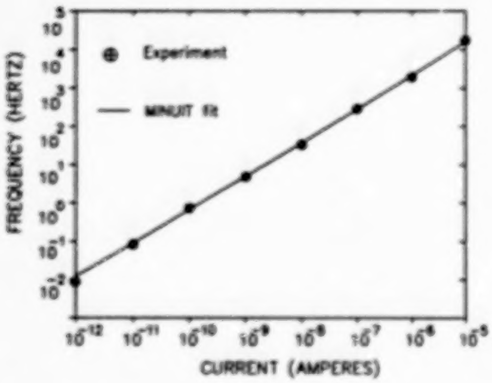


Figure 5: Experimental data points [6] and the calculated (MINUIT) fit. The output dynamic range is slightly less than the input dynamic range corresponding to sublinear current-to-frequency conversion. The model gives an extremely good fit which ranges across 7 decades.

tion of the Cornsweet algorithm is an unusually interesting area of research because the relevant retinal and neural mechanisms have not yet been identified.[1] On the other hand, a great deal is known about the connectivity of the retinal neural network, so that biological plausibility might be invoked as a broad, qualitative constraint on network architecture. For applications, it is of course not necessary to be unduly constrained by biological analogies and differences will surely appear in a silicon device approach. However, a point which is frequently made in connection with neural network research is that, at our present level of understanding, there is probably much to be gained from a reverse engineering analysis of high performance biological systems.

2 Devices For Parallel Asynchronous Processing

Previous studies[6] of current driven p^+-n-n^+ diodes led to the discovery of input current dynamic ranges up to 10^7. The corresponding output pulse rate range was sometimes less than the input range. See Fig.5. We have developed a model for spontaneous firing during current-to-frequency conversion (I-to-f conversion) and used the model to analyze the data shown in Fig. 5 using a program developed at CERN called MINUIT[14]. A key feature which is explained is that the slope of lnf vs lnI is not always unity. The data in Fig. 5 correspond to $f \propto I^{1-\epsilon}$. A simple picture with an equal amount of charge transfer in each impulse would explain $f \propto I$. However, more detailed device modeling was required to understand sublinear $f \propto I^{1-\epsilon}$ behavior.

2.1 Sensors and Sensor-Processor Interfacing

This section describes experimental work on sensors and sensor-processor interfacing. Results have been obtained for reverse biased p-i-n photodiodes which are useful in the visible, ultraviolet and near infrared regions and for infrared detectors which are useful in the far infrared region. The most dramatic results in terms of dynamic range came from visible light measurements with reverse biased p-i-n photodiodes where the dark current reduction associated with cooling led to the enormous dynamic range shown in Fig. 6.

---

[1] T. Cornsweet, private communication.

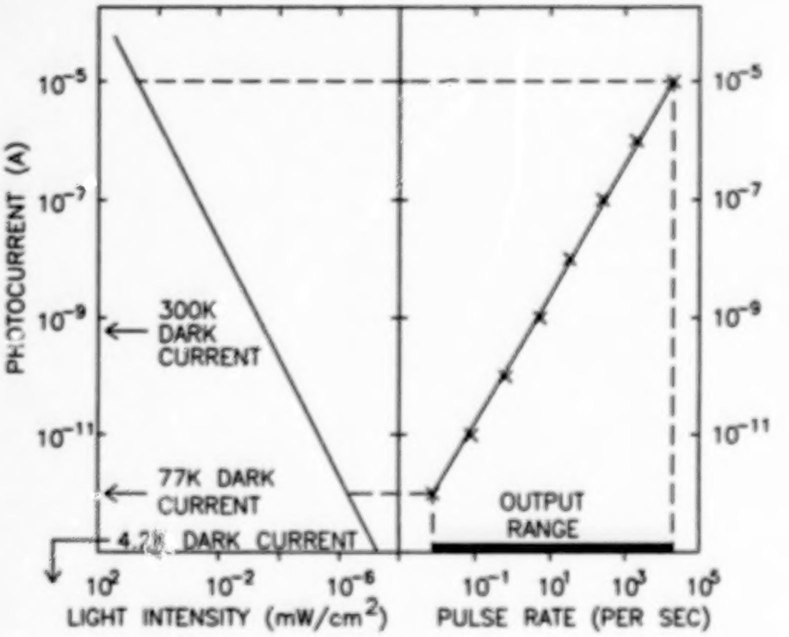


Figure 6: The graphs show that the photocurrent output of a reversed biased p-i-n photodiode (left graph) in response to visible light overlaps the large dynamic range of p^+-n-n^+ devices (right graph). This implies that such photodiodes could be directly interfaced to a parallel asynchronous processor based on p^+-n-n^+ devices. Such interfacing would preserve the high dynamic range.

2.2 Two-Dimensional Data Transfer

We have performed experiments in order to examine means of 2-D parallel data transfer without multiplexing. The idea is that neuronlike spiketrains could be used to drive arrays of LEDs for 2-D optical data transfer. The LED firing pattern could be recorded using a video camera or received by a photodiode array.

In our experiments, a p^+-n-n^+ diode drives a LED which is also inside a cryogenic environment. When a pulse from the p^+-n-n^+ diode goes above the threshold voltage of the LED, the LED starts conducting and emitting light. The LED inside the dewar can be viewed from outside the dewar. This is convenient and avoids a heat load. While the p^+-n-n^+ diode pulse is greater than the threshold the LED will be on. The pulse will decay according to the circuit parameters, i.e., the time constant. The speed of data transfer will be limited by the RC time constant. Optimal performance corresponds to dissipation of power in the LED rather than in the load resistor so that the RC decay is undesirable from the point of view of power considerations as well as avoidance of pile up at high pulse rates.

2.3 Ultralow Power Requirements

Massive processing tasks, operation in space and cooling for high performance (low dark current) operation are all factors which point to the benefits of low power operation. Von Neumann's estimate of the power consumption of the brain was 10-25 watts [15] which is remarkably small for a system with $\sim 10^{11}$ neurons, i.e. ~ 100 picowatts/neuron. It has been argued that arrays of small p^+-n-n^+ diodes could offer comparably low (or even lower) power consumption.[5] Scaling down the device size will scale down the power requirements per device. For p^+-n-n^+ diodes, we have observed pulses with energy dissipation down to 4 picojoules/mm^2/pulse and a quiescent power dissipation of 10 picowatts/mm^2. Considering the thermodynamic efficiency of cooling, these numbers correspond to 290 picojoules/mm^2/pulse and 710 picowatts/mm^2 at room temperature. For comparison, we note that the retina chip of Mead and Mahowald[16] has a power dissipation of 4 microwatts/mm^2.

The range of p^+-n-n^+ diode action potential pulse heights observed to date is from about 20 millivolts to 50 volts with the low end of the range corresponding to the low power figures reported here. The low end of this range of pulse heights is comparable to action potential pulse heights of real neurons. Our device physics modeling of spiketrain generation could lead to further power reductions if necessary. Low power dissipation would permit substantial processing to be performed at or just behind a focal plane array of detectors which are normally cooled to achieve high performance.

This electronic approach is remarkably well suited to neural network emulation and parallel asynchronous processing. Such hardware offers the possibility of 2-D parallel image processing in conjunction with image acquisition in much the same way as image acquisition and early processing are performed in natural vision systems. This is of interest because it is generally acknowledged that many image processing tasks are performed by natural vision systems with noteworthy speed, even in comparison with the fastest available systems employing conventional electronics.

We have identified certain image processing algorithms (IDS and pyramid) as being (A) especially well suited to our 2-D parallel approach and (B) of special relevance to potential NASA applications. The ultimate system which could emerge from our research would be a real time, high resolution, high dynamic range, low power integrated (single package) focal plane array-2-D parallel processor. The processor would be hard-wired to implement particular algorithms. Successive processing levels could perform a succession of processing tasks. For example, one might want to perform further parallel processing on the output of an IDS algorithm stage.

2.4 Parallel Processing Speed and Data Compression

In a fully parallel processing system the bandwidth per processing channel can be of the order of the bandwidth required per pixel. By contrast, serial systems introduce bottlenecks and require higher processing speeds which scale with the array size.

Standard planar semiconductor technology dictates that signals be transmitted to the edge of chips where the 2-dimensional input image data flow confronts a 1-D perimeter bottleneck. The number of detectors per preamplifier is a measure of the chip level bottleneck. Conventional approaches thus require increased electronics to reduce bottlenecks. The devices discussed here require **no preamplifiers** so we are able to go all the way to 2-D parallel processing and eliminate bottlenecks without introducing a 2-D array of preamplifierss or even one preamplifier. For conventional systems, there would be higher power requirements proportional to the number of preamplifiers. This would be disadvantageous in a cryogenic environment especially in NASA space applications. (Note that conventional CCD cameras are cooled to achieve their best performance.)

The dimensionality of desired patterns at the output plane of a 2-D signal processor provide a rough qualitative measure of the degree of data compression which is possible:

Detected Features	*Input → Output Dimensionality*
Set of point targets	2D → 0D
Set of edges	2D → 1D
Full optical data flow	2D → 2D

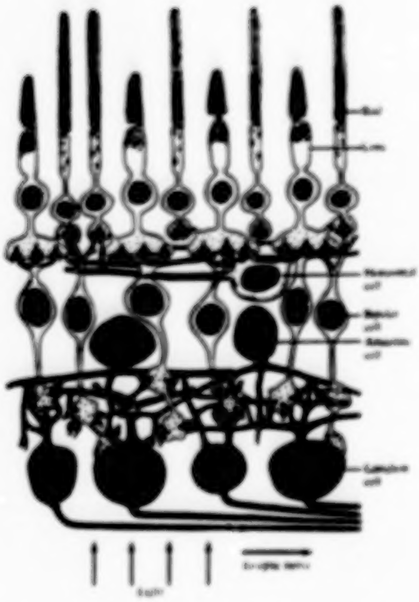


Figure 7: Schematic representation of the primate retina.[17] The light sensitive rods and cones pass signals to horizontal cells where lateral inhibition may occur. Signals then pass through bipolar cells to the highly interconnected plexiform layer. From the plexiform layer, signals are transmitted to the ganglion cells which connect to fibers of the optic nerve. Reproduced with the permission of Sinauer Assoc. Inc.

The relevant output from the IDS algorithm conveys information in the neighborhood of an edge, so it's not strictly one-dimensional. Besides edge contour information, there is additional information in the IDS algorithm output, namely, intensity-ratios or reflectance-ratios associated with the two regions on either side of the edge. If we assign one normalized reflectance to each 2-D subregion (plaquette) within a closed contour revealed by edge detection, then the IDS output data could be compressed into a "sketch" displaying 1-D edge contours (plaquette perimeters) and a set numbers (normalized reflectances), one for each 2-D plaquette.

2.5 Similarity with Image Processing in Natural Vision Systems

Our parallel asynchronous processing strategy, our neuronlike information coding and our intrinsically 2-D data flow all suggest a close analogy with natural vision systems. In addition, our approach preserves the geometrical relationship of neighboring channels as is the case in natural vision systems. A key aspect of processing in natural vision systems is lateral interaction between neighboring or nearby processing channels. It thus appears that our hardware approach is well suited to implementation of image processing schemes which parallel those of natural vision systems.

Lateral interaction between nearby processing channels is associated with vision system spatial filtering. Lateral interactions determine the receptive fields of neuron processing elements and the point-spread functions of individual photoreceptors. In natural vision systems, neurons mediate lateral interactions as shown in Fig. 7. In the retina chip of Mead and Mahowald, lateral interactions are incorporated via a resistive network.[16] However, no spiketrain generation and no intensity-to-frequency conversion occur as in the retinas of natural vision systems. See Fig. 8.

In our approach, neuronlike spiketrain generation is used. An artificial neuron circuit and the analogy with real, stereotypical neurons is illustrated in Fig. 9.[5]

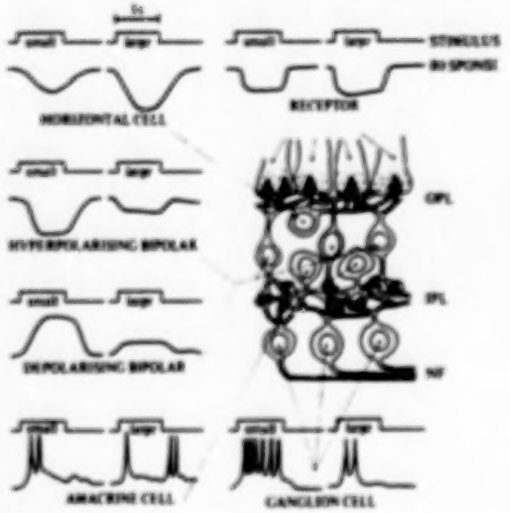

1s
small
large
STIMULUS
RESPONSE
HORIZONTAL CELL
RECEPTOR
OPL
HYPERPOLARISING BIPOLAR
IPL
DEPOLARISING BIPOLAR
NF
AMACRINE CELL
GANGLION CELL

Figure 8: Waveforms recorded from various cells in the vertebrate retina when a small stimulus spot is shown on retina photoreceptors, and when a large spot that includes surrounding elements is used. The stimuli last about 1 second, and the responses are up to 30 mV in amplitude. OPL and IPL refer to outer and inner plexiform layers and NF refers to the nerve fibers. [18] Reproduced with the permission of Cambridge University Press.

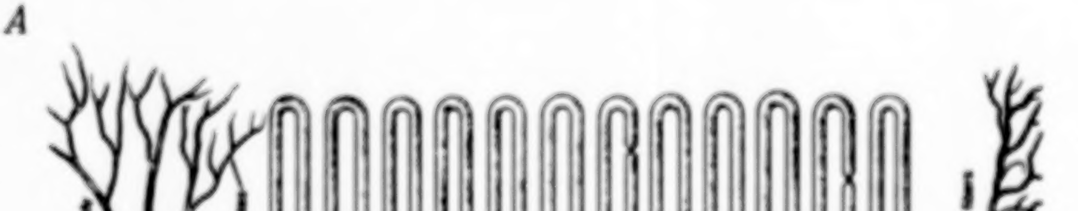

A
B
INPUTS
V_{BE}
C_1
R
V
C
R
TRANSMISSION LINE
R_L
OUTPUTS
C_1

Figure 9: *A*): Features of a typical neuron from Kandel and Schwartz [19] and *B*): our artificial neuron, which exhibits the summation over synaptic inputs and fan-out. The input and output capacitive couplings are useful in conjunction with spiketrains. The darkened diode is a p-n junction device used for pulse height discrimination. The other diode is a p^+-n-n^+ diode used for spiketrain generation.

2.6 Hardware Implementation of the IDS Image Processing Algorithm

Our approach to circuit design and hardware implementation is guided in part by the fact that the IDS algorithm was intended to be in accord with several key features of natural vision systems. Thus, the algorithm or a close approximation to the algorithm is being implemented by biological neural networks whose general structure is known. See Figs. 7 and 8. On the other hand, no direct link between the IDS algorithm and retinal neural network architecture yet exists. Understanding this link would be of direct significance to circuit designs for parallel asynchronous neuronlike implementation of the IDS algorithm and, in addition, would be of significance in the field of vision.

A hardware implementation of the IDS algorithm need not have an exact retinal neural network analog. Therefore, a clear-cut conceptual advance in relation to retinal implementation, although desirable, is not a necessary condition for IDS hardware implementation. However, even without detailed understanding of retinal processing, circuit design efforts can benefit from knowledge of the general features of retinal neural network architecture, such as those apparent in Figs. 7 and 8.

To illustrate our approach, we show in Fig. 10 a preliminary strategy for implementation of the IDS algorithm which possesses some architectural similarity to retinas.

A key feature of the implementation concept is a 2-D array of constant current sources, in one-to-one correspondence with the photodetectors. The lateral spreading of this current is associated with the IDS point-spread function. Pulses associated with the spiketrain coding of the photodetector outputs gate the forward flow of current from the current sources. High intensities provide more rapid gating and more forward current flow which competes with and limits lateral spreading of the current. Thus, higher intensities diminish lateral spreading as in the IDS algorithm. On the other hand, the constancy of each current source and current conservation during spreading produce a constraint that the integrated output current must also be constant, despite its intensity dependent spreading. This is analogous to the IDS constant "volume" constraint, i.e. constant intensity × area.[9]

A key aspect of the circuit involves capacitive couplings as in Fig. 9 which permit information transfer, but no net time-averaged current flow, i.e. no dc component. This permits light intensity to play a role but the photocurrent does not add to the dc current coming from the constant current sources.

The output is again coded into spiketrains for further processing or for output typically via an LED array. Note that retinal outputs to the brain are coded into spiketrains by the ganglion cells.

The implementation concepts described here are preliminary concepts. It is very likely that further considerations will be needed to produce quantitative agreement with the intensity dependence of IDS spatial scaling associated with the point-spread function from the input point (x,y) to the output point (p,q)

$$I(x,y) \times S\left[I(x,y) \times \left((x-p)^2 + (y-q)^2\right)\right] \tag{1}$$

where the non-negative real function S is normalized by:

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} S(p^2 + q^2)\,dp\,dq = 1 \tag{2}$$

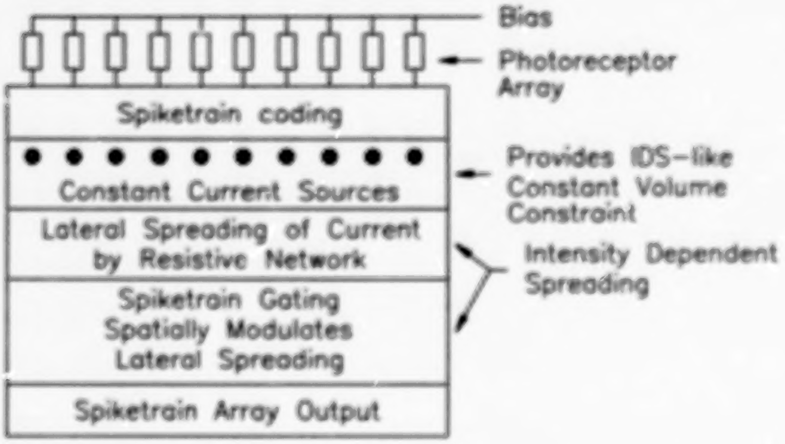

Bias
Photoreceptor Array
Spiketrain coding
Constant Current Sources
Provides IDS-like Constant Volume Constraint
Lateral Spreading of Current by Resistive Network
Intensity Dependent Spreading
Spiketrain Gating Spatially Modulates Lateral Spreading
Spiketrain Array Output

Figure 10: Functional description of a preliminary circuit concept for implementation of the IDS algorithm.

Pixel or detector size and the spatial sampling period (or frequency) are important issues in connection with fundamental image processing. This idea has been emphasized in the work of Huck, Fales, McCormick, Park, Halyo, Samms and Stacy.[20,21,22] The issue is not dealt with in the continuum formulation of the IDS algorithm.[9]

Furthermore, there is an issue with respect to circuit architecture which is similar to an issue raised by Cornsweet[2] in connection with retinal implementation. This concerns implementing IDS point-spread functions (one for each input) on a shared network structure. For linear point-spread functions, this is easy to envisage. However, with the nonlinear intensity dependence of the IDS algorithm, one worries that nonlinear spreading associated with one photoreceptor will interfere with the spreading associated with another photoreceptor if spreading occurs over a shared network. Non-shared spreading networks would solve this problem but would be more complex (higher parts count) and would contradict the impression that retinal neural networks (as shown in Figs. 7 and 8) are shared. This issue deserves further study.

3 Acknowledgment

This work was supported in part by the NASA under contract No. NAS1-18850. The authors are pleased to acknowledge F. Huck and his colleagues for discussion of issues involved with application of image processing algorithms. The authors are also pleased to acknowledge helpful communication with T. Cornsweet, E. Kurrasch and G. Westrom.

References

[1] D. D. Coon and A. G. U. Perera, Solid-State Electronics **29**, 929 (1986).

[2] D. D. Coon and A. G. U. Perera, Int. J. Electronics **63**, 61 (1987).

[3] D. D. Coon and A. G. U. Perera, Solid State Electronics **31**, 851 (1988).

[4] D. D. Coon, S. N. Ma and A. G. U. Perera, Phys. Rev. Lett. **58**, 1139 (1987).

[5] D. D. Coon and A. G. U. Perera, *New Hardware for Massive Neural Networks* in **Neural Information Processing Systems**, American Institute of Physics, 1988, pp. 201–210.

---

[2] T. Cornsweet, private communication.

[6] K. M. S. V. Bandara, D.D. Coon and R. P. G. Karunasiri, Appl. Phys. Lett. **51**, 961 (1987).

[7] D. Marr and E. Hildreth, Proc. R. Soc. Lond. B **207**, 187-217 (1980).

[8] D. Marr, *Vision*, W. H. Freeman and Co., San Fransisco, 1982.

[9] T. N. Cornsweet and J. I. Yellott, Jr., J. Opt. Soc. Am. A **2**, 1769 (1985).

[10] T. N. Cornsweet and J. I. Yellott, Jr., J. Opt. Soc. Am. A **3**, 165 (1986), errata.

[11] T. N. Cornsweet, *Visual Perception*, Academic Press, Inc., New York, 1970.

[12] J. I. Yellott, Jr., J. Opt. Soc. Am. A **4**, (1987).

[13] K. M. S. V. Bandara, D. D. Coon and R. P. G. Karunasiri, Opt. Eng. 27, 471 (1988).

[14] F. James and M. Roos, Computer Physics Communications **10**, 343 (1975).

[15] J. von Neumann, *The General and Logical Theory of Automata* in Collected Works Vol. 5, A. H. Taub, ed. Pergamon Press, New York, 1961.

[16] C. A. Mead and M. A. Mahowald, Neural Networks **1**, 91 (1988).

[17] S. W. Kuffler, J. G. Nicholls, and A. R. Martin, *From Neuron to Brain*, Sinauer Associates Inc., Massachusetts, 1984, page 125.

[18] H. B. Barlow and J. D. Mollon (Eds.), *The Senses*, Cambridge University Press, 1982.

[19] E. R. Kandel and J. H. Schwartz, *Principles of Neural Science*, Elsevier, New York, 1985, page 15, Reproduced by permission of Elsevier Science Publishing Co., N.Y.

[20] F. O. Huck, C. L. Fales, N. Halyo, R. W. Samms and K. Stacey, J. Opt. Soc. Am. A **2**, 1644 (1985).

[21] C. L. Fales, F. O. Huck, J. A. McCormick and S. K. Park, J. Opt. Soc. Am. A **5**, 300 (1988).

[22] F. O. Huck, C. L. Fales, J. A. McCormick and S. K. Park, J. Opt. Soc. Am. A **5**, 285 (1988).

Neural Networks for Data Compression and Invariant Image Recognition

Sheldon Gardner
Naval Research Laboratory
Washington, DC

SUMMARY

An approach to invariant image recognition [I^2R], based upon a model of biological vision in the mammalian visual system [MVS], is described. The complete I^2R model incorporates several biologically inspired features: exponential mapping of retinal images, Gabor spatial filtering, and a neural network associative memory. In the I^2R model, exponentially mapped retinal images are filtered by a hierarchical set of Gabor spatial filters [GSF] which provide compression of the information contained within a pixel-based image. A neural network associative memory [AM] is used to process the GSF coded images. We describe a 1-D shape function method for coding of scale and rotationally invariant shape information. This method reduces image shape information to a periodic waveform suitable for coding as an input vector to a neural network AM. The shape function method is suitable for near term applications on conventional computing architectures equipped with VLSI FFT chips to provide a rapid image search capability.

INTRODUCTION

Neural networks offer a potential for technology innovation to provide the next generation of on-board processing [OBP] capability in space-based systems for strategic defense and surveillance as well as other non-military space applications such as remote sensing of the environment. The data collection capabilities of space-based imaging sensors are expected to continue to improve dramatically, further outstripping the ability of operators to exploit image data in real time. One of the goals of the Image Processing Research [IPR] Program at the NRL Naval Center for Space Technology is to develop applications for neural network-based invariant image recognition [I^2R][1-4].

The encoding of images by the mammalian visual system [MVS] is a subject which has challenged vision researchers for centuries. In the past several years significant progress has been made by Daugman and others towards an understanding of how images are processed within the MVS [5-12]. The basic architecture for invariant image recognition is shown in Figure 1. We assume that the MVS performs a sequence of space and space-time mappings which we call scale-space transformations [SST] [1,2]. The first SST to occur in the MVS is a logarithmic spatial mapping which occurs in the retina in the vicinity of the fovea. This mapping, which we call the LZ-SST, produces scale and rotational invariance in the foveal image [14,15]. A second SST, which we call the cortical filter SST, or CF-SST, occurs throughout the lateral geniculate nucleus and the striate cortex. The function of the CF-SST is to provide a coded representation of the image for associative memory processing which takes place in higher cortical areas. We have suggested that, among other operations, the CF-SST includes a hierarchical network of Gabor filters to map the retinal image into a four-dimensional function of two spatial variables and two spatial-frequency variables. Functionally, this mapping is equivalant to computation of the 4-D Cross-Wigner Distribution [CWD][1,12,13]. These complex spatial filtering operations occur within the the second block shown at the top of Figure 1. The encoded image features are then processed by the neural network associative memory [AM] as shown in the third block of Figure 1.

In the next section we describe the shape function method for coding of scale and rotationally invariant shape information into a scalar waveform. This method can reduce line object shape information to a scalar waveform suitable for processing by a VLSI FFT array or for coding as an input vector to a neural network AM.

CODING OF SHAPE FUNCTIONS

Motivated by the properties of the MVS, we can represent a static image by means of a hierarchical relational graph [HRG][4]. At each level of the hierarchy, we constructed a set of nodes (simple objects), and a relational graph (complex object) based upon the relations between the nodes. At the next lowest level in the hierarchy (finer resolution), each node is treated as a complex object, composed of its own set of connected simple objects. Although, we describe the HRG structure in a top-down manner, in the MVS data flow actually takes place in a bottom-up manner, since image information is first processed in the visual cortex, then sent to higher areas of the brain, such as the cerebral cortex. Recognition of a face can be used as a simple example of this process. Starting with the placement of features (e.g. eyes, nose, etc.) we recognize a face as a complex object composed of simple objects (features). On the next hierarchical level we examine individual facial features. Fig. 2 illustrates the hierarchical representation of object shape. The complex object $F1[\cdot]$, shown in Figure 2, can be represented in terms of a three-level hierarchical notation $F_1[G_1[H_1], G_2[H_2]]$.

Figure 3 illustrates a two-step process which can be used to obtain the shape features of a broad-band multi-level image. The nonlinear trace operation shown in Figure 3(b) converts a bit-mapped image into a set of objects. An example of this type of trace operation can be found in commercial microcomputer software (e.g. Digital Darkroom®).

Shape information can be used in the construction of object features vectors useful for object recognition. We illustrate how, after posterization and tracing between fixed grey levels, shape information can be coded into a scalar shape function which characterizes a line object. For high speed applications which require special purpose hardware, such as VLSI array processors implementing FFT algorithms, these shape functions can be processed with conventional computers (e.g a Hypercube® or a Connection Machine®). In the future, when massively parallel neural network computers become available, shape functions can be coded into feature vectors for input to a neural network AM.

As an illustration of the shape function process, an aircraft line object is shown in Figure 4(a) together with the corresponding shape function shown in Figure 4(b). To compute the shape function, we first select a suitable centroid within the object boundary. The shape function is then defined as the distance from this centroid to the object contour measured as a function of distance around the object perimeter. Figure 4(b) is a plot of the aircraft shape function measured from the nose (top). Individual features, such as the engines, can be clearly identified. Figures 5 and 6 show line objects and shape functions for two other aircraft of different types. Figures 7 and 8 show the data for two of the aircraft with a 10 db S/N. The identifying features of each aircraft are still clearly visible in the shape functions. In practice, a sequence of noisy images will usually be available for processing. If the spatial noise background between images in the sequence is uncorrelated, an improvement in S/N will occur when averaging over multiple frames.

CONCLUDING REMARKS

A model for invariant image recognition, based on the properties of the MVS, has been described. The model includes a hierarchical representation of shape information for complex objects. Each level in the hierarchy is represented by a collection of line objects. Through a nonlinear tracing operation the pixel image of each objects is converted to a shape contour. This contour is then represented by a scalar shape function defined as the distance from a centroid within the object to the contour expressed as a function of distance around the object perimeter. This scalar shape waveform uniquely represents object features and can be processed with conventional FFT hardware. Simulations are used to demonstrate the viability of the approach.

REFERENCES

1. S. Gardner, Morphology of Neural Networks in the Mammalian Visual System, First Annual Meeting of the International Neural Network Society, Boston, Mass., Sept. 1988.

2. S. Gardner, Ultradiffusion, Scale Space Transformation, and the Morphology of Neural Networks, IEEE Annual International Conference on Neural Networks, San Diego, Cal., July 1988.

3. S.Gardner, Statistical Dynamics of Ultradiffusion in Hierarchical Systems, Proceedings of the IEEE First International Conference on Neural Networks, M. Caudill, C. Butler, Eds., IEEETH00191-7, 1987.

4. S. Gardner, An Approach to Multi-Sensor Data Fusion based upon Complex Object Representations, Proceedings of the 1988 Symposium on Command and Control Research, U.S. Navy PG School, Monterery, Cal., June 1988 .

5. J. Daugman., Two-Dimensional Spectral Analysis of Cortical Receptive Field Profiles, Vision Research, Vol.20, p. 847, 1979.

6. J. Daugman, Uncertainty Relation for Resolution in Space, Spatial Frequency, and Orientation Optimized by Two-Dimensional Visual Cortical Filters, J. Optical Society of America A., Vol. 2, No. 7, 1985.

7. J. Daugman, Kammen, D., Image Statistics, Gases, and Visual Neural Primitives, Proceedings of the IEEE First International Conference on Neural Networks, M. Caudill, C. Butler, Eds. IEEETH00191-7, 1987.

8. J. Daugman., Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression, IEEE Trans. ASSP, Vol. 36, No. 7, p. 1169, July 1988.

9. B. Richmond, L. Optican, M. Podell, H.Spitzer, Temporal Encoding of Two-Dimensional Patterns by Single Units in Primate Inferior Temporal Cortex, I. Response Characteristics, J. of Neurophysiology, 57, No. 1,p. 132, Jan. 1987.

10. B. Richmond, L. Optican, Temporal Encoding of Two-Dimensional Patterns by Single Units in Primate Inferior Temporal Cortex, II. Quantification of Response Waveform, J. of Neurophysiology, 57, No. 1,p. 147, Jan. 1987.

11. B. Richmond, L. Optican, Temporal Encoding of Two-Dimensional Patterns by Single Units in Primate Inferior Temporal Cortex, III. Information Theoretic Analysis, J. of Neurophysiology, 57, No. 1,p. 162, Jan. 1987.

12. L. Jacobson, H. Wechsler, Joint Spatial/Spatial Frequency Representation, Signal Processing 14, p.37-68, North Holland, 1988.

13. D. Gabor, Theory of Communication, J.IEE, Vol.93, p.429, 1946.

14. Messner, R.,(1984), Smart Visual Sensors for Real-Time Image Processing and Pattern Recognition based upon the Human Visual System. Phd. Dissertation, Clarkson University, Potsdam, N.Y.

15. Messner, R., Szu, H. (1985), An Image Processing Architecture for Real Time Generation of Scale and Rotation Invariant Patterns, CVGIP, Vol.31, p.50-66.

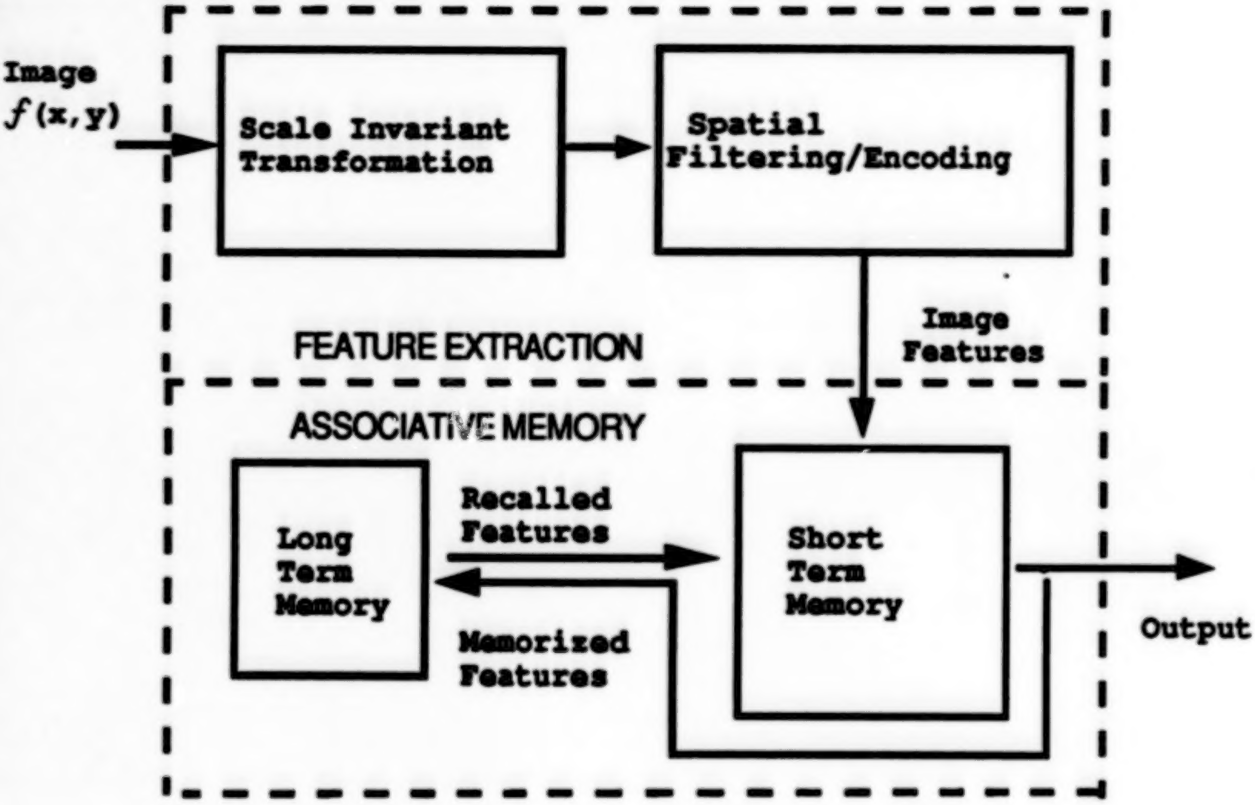


Image
$f(x,y)$
Scale Invariant Transformation
Spatial Filtering/Encoding
Image Features
FEATURE EXTRACTION
ASSOCIATIVE MEMORY
Long Term Memory
Recalled Features
Memorized Features
Short Term Memory
Output

Figure 1. Architecture for invariant image recognition.

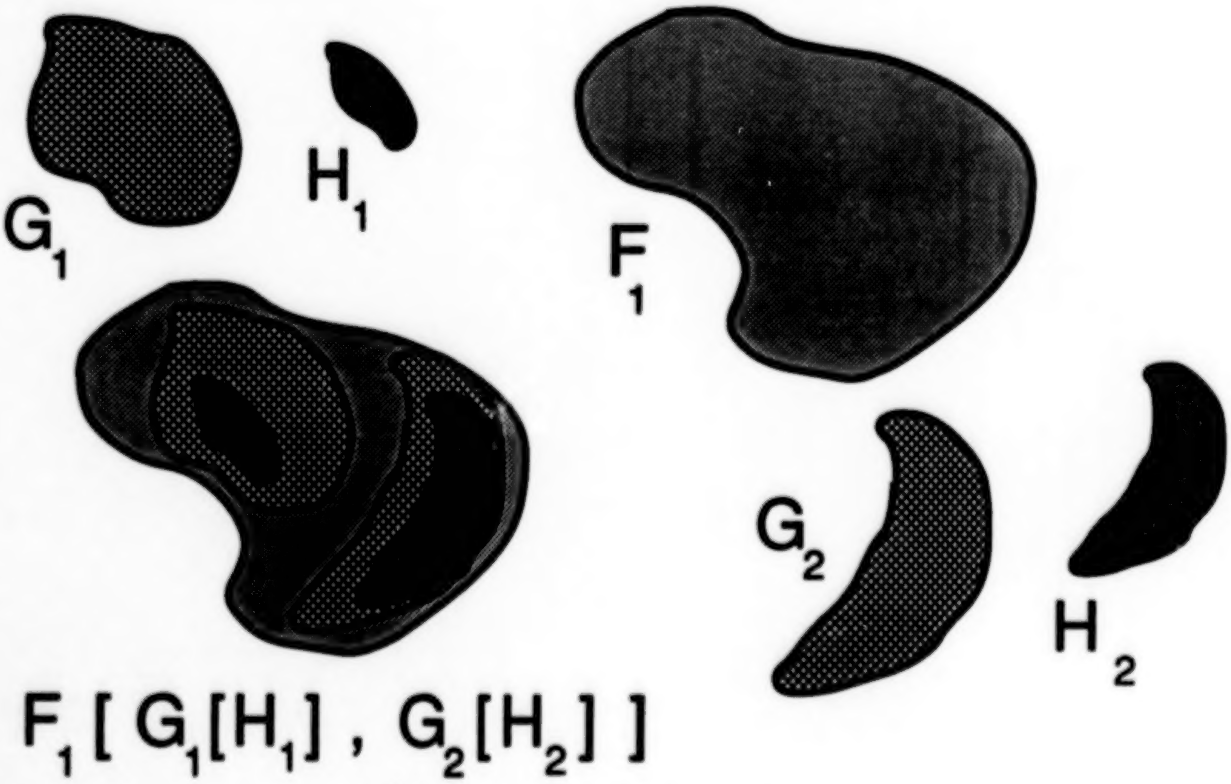

G1
H1
F1
G2
H2
F1 [G1[H1] , G2[H2]]

Figure 2. A hierarchical representation for object shape

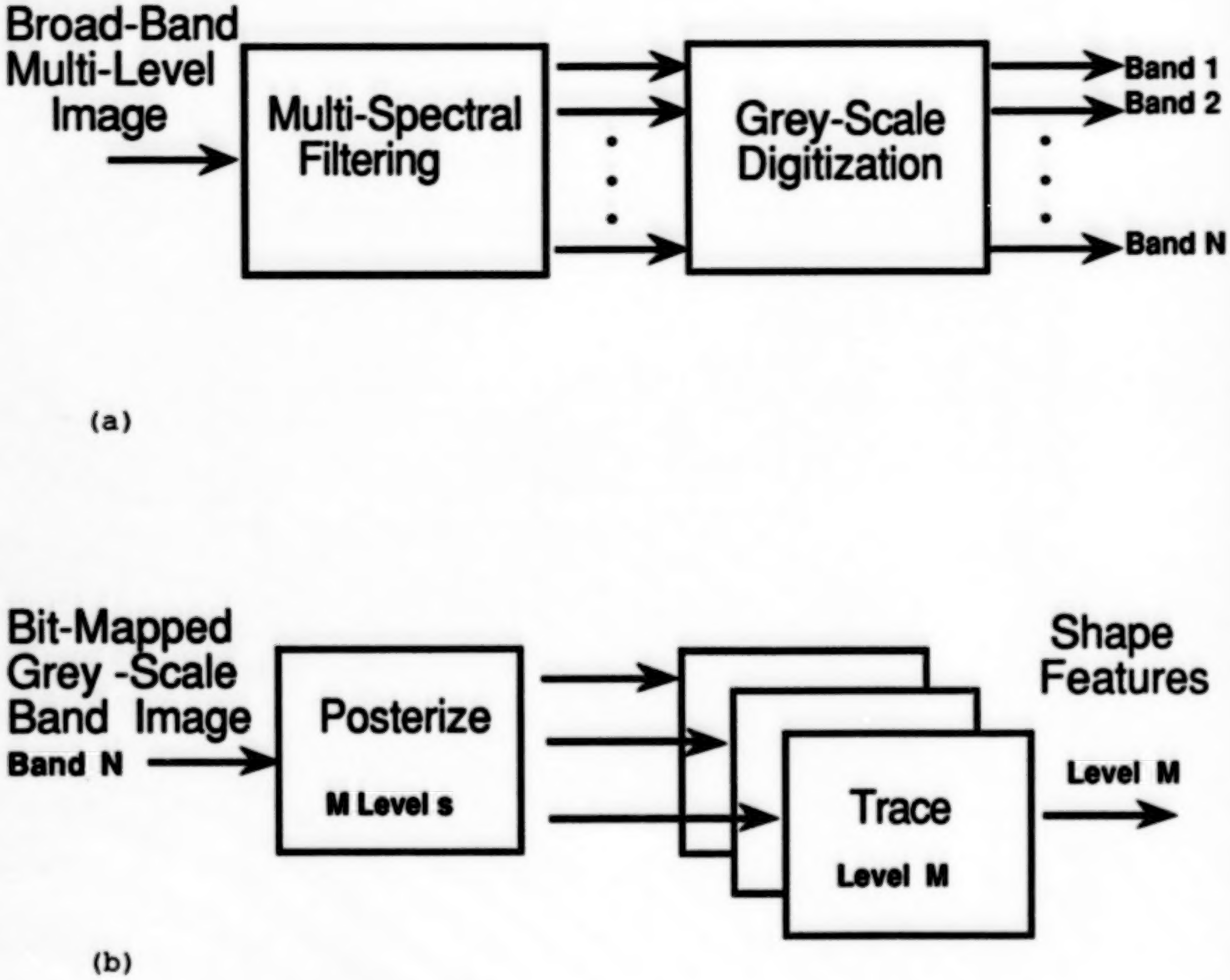

Broad-Band
Multi-Level
Image
Multi-Spectral
Filtering
Grey-Scale
Digitization
Band 1
Band 2
Band N
(a)
Bit-Mapped
Grey -Scale
Band Image
Band N
Posterize
M Level s
Trace
Level M
Shape
Features
Level M
(b)

Figure 3. Steps in obtaining shape features from a broad-band image.

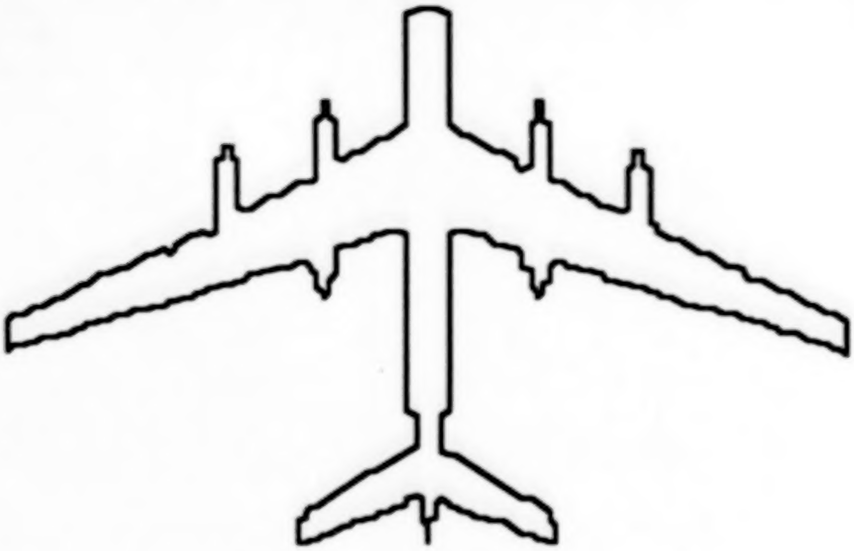


Figure 4(a) Aircraft 1 line object

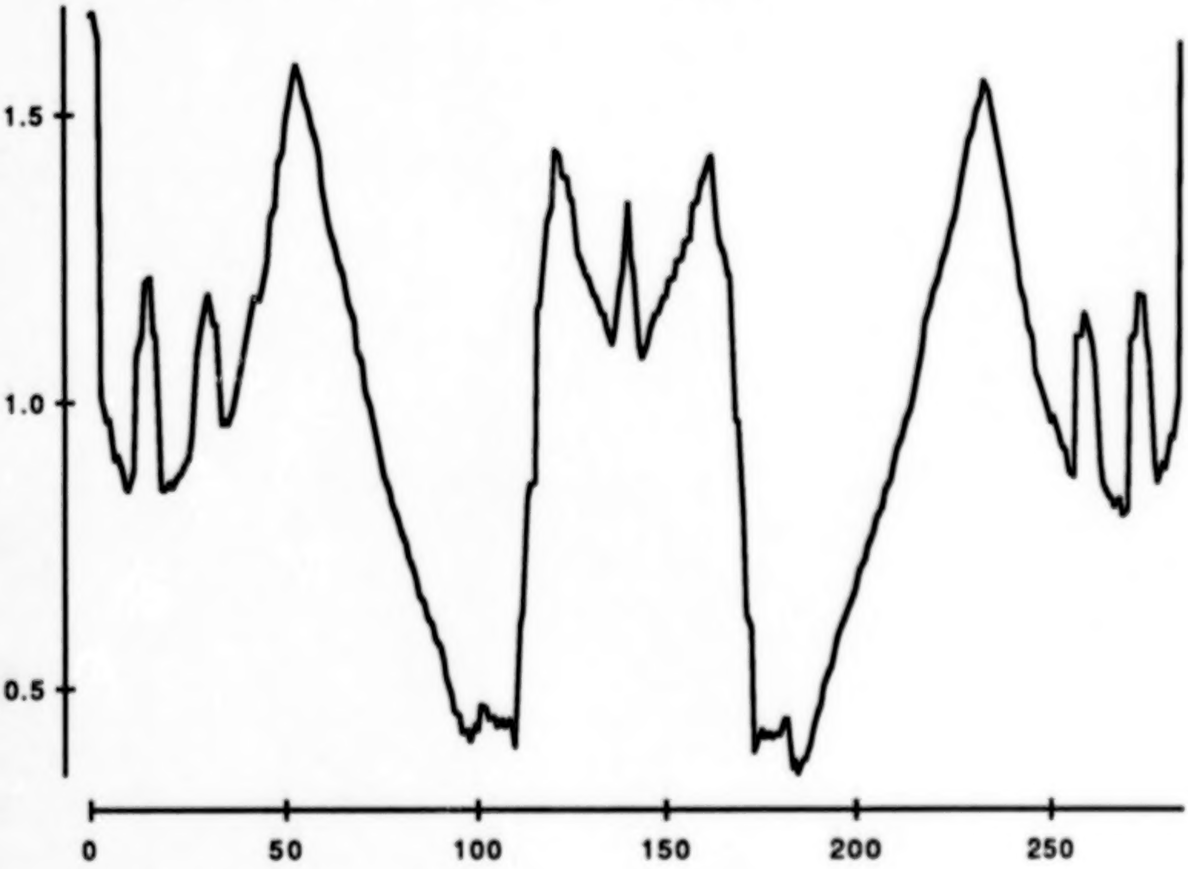


Figure 4(b) Aircraft 1 shape function.

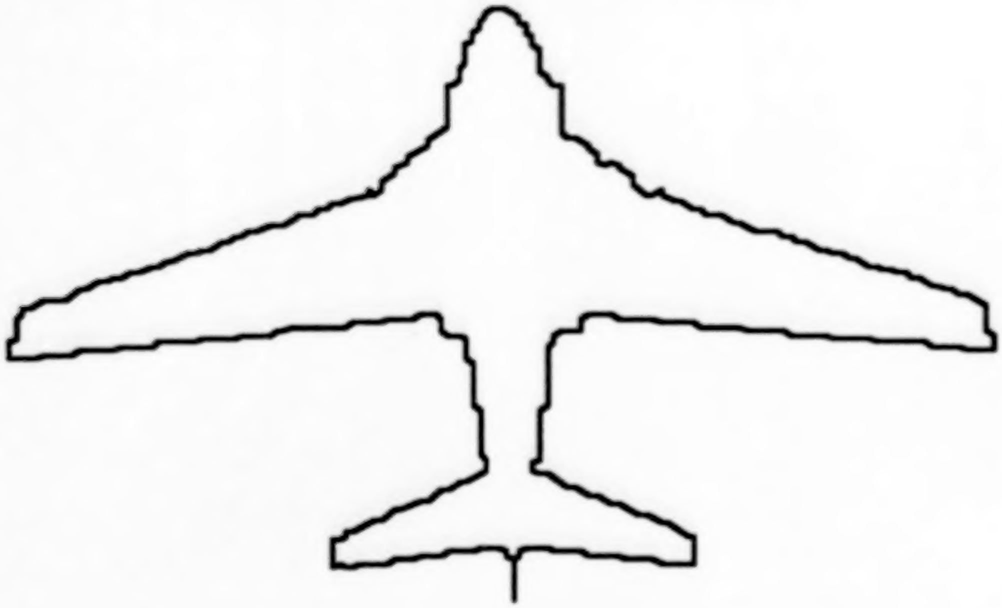


Figure 5(a) Aircraft 2 line object

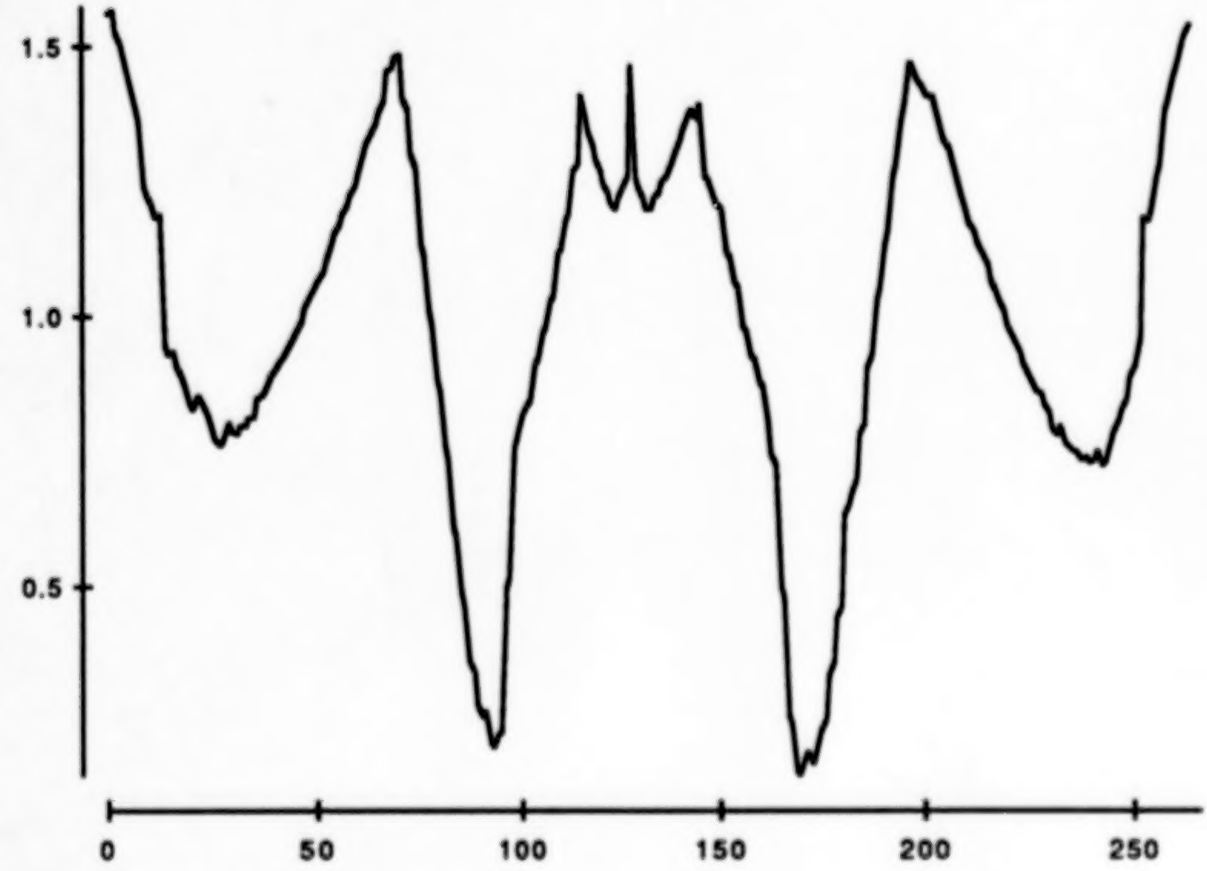


Figure 5(b) Aircraft 2 shape function.

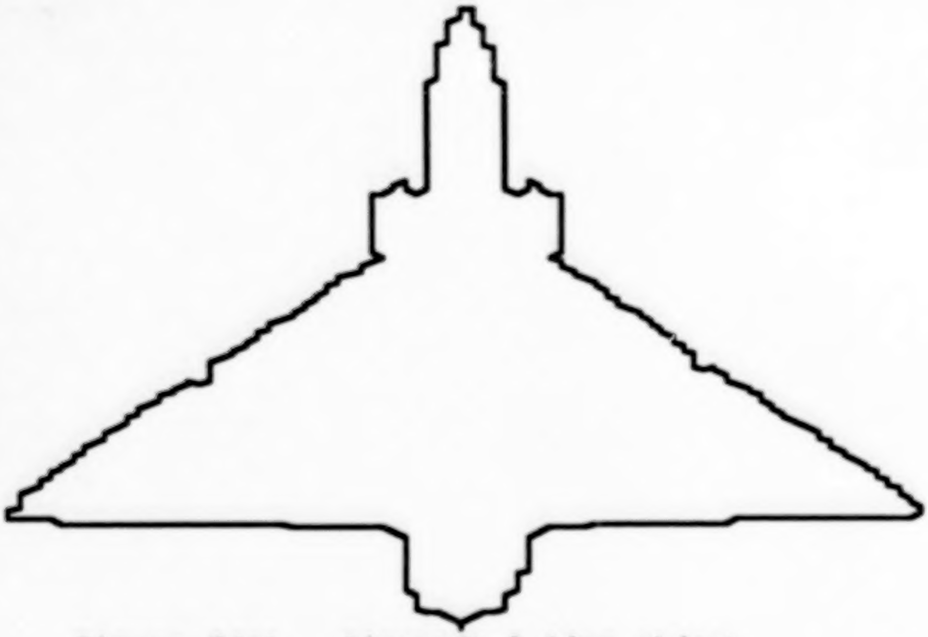


Figure 6(a) Aircraft 3 line object

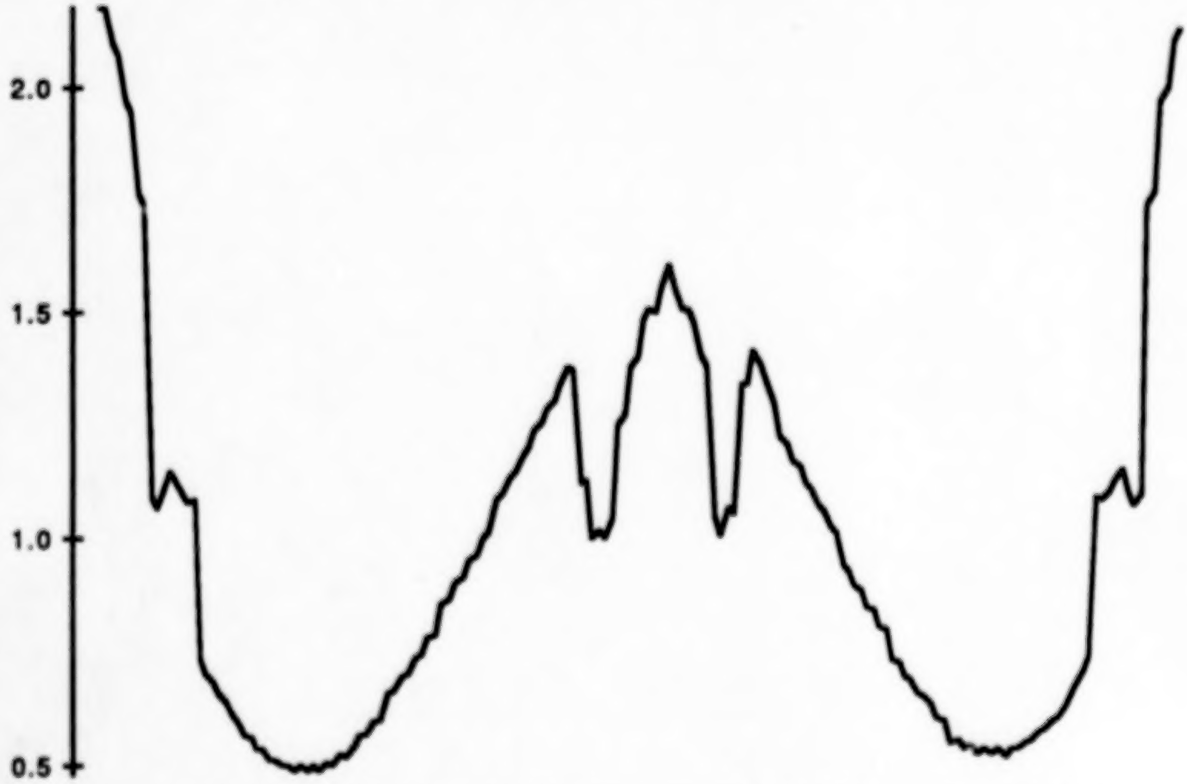


Figure 6(b) Aircraft 3 shape function.

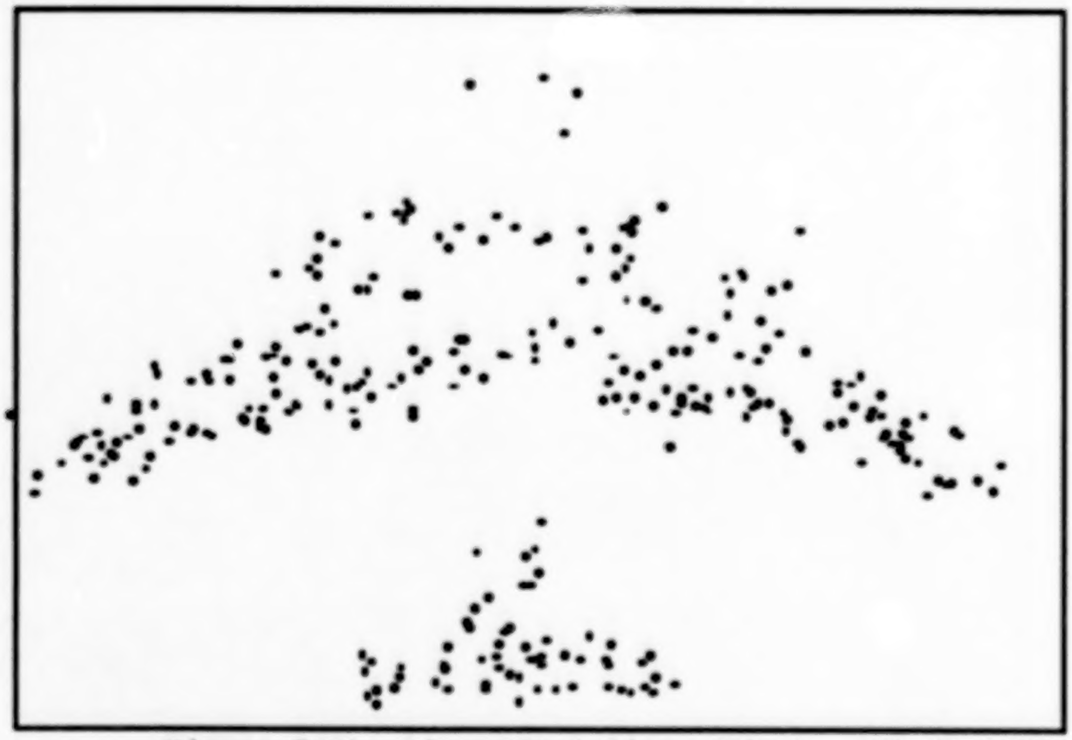


Figure 7(a) Aircraft 1 line object (10 db S/N)

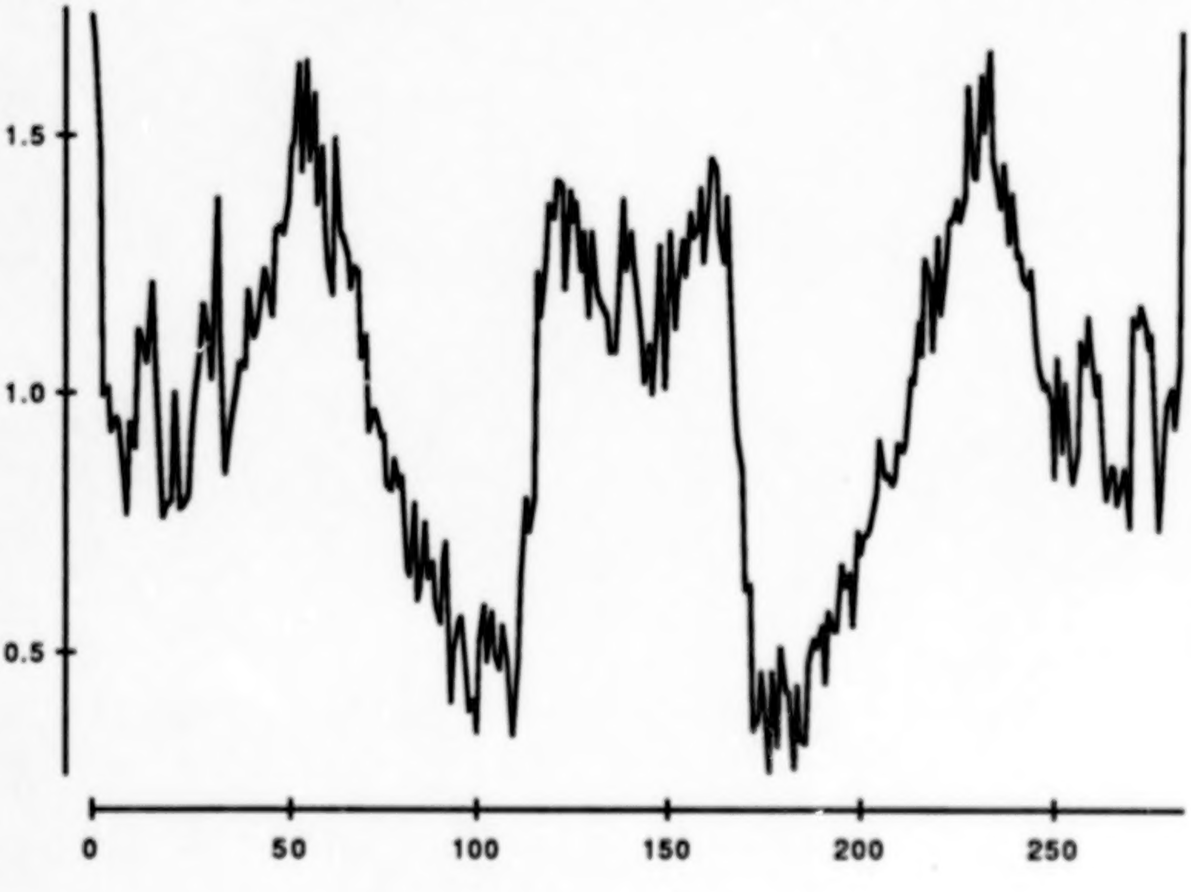


Figure 7(b) Aircraft 1 shape function (10 db S/N)

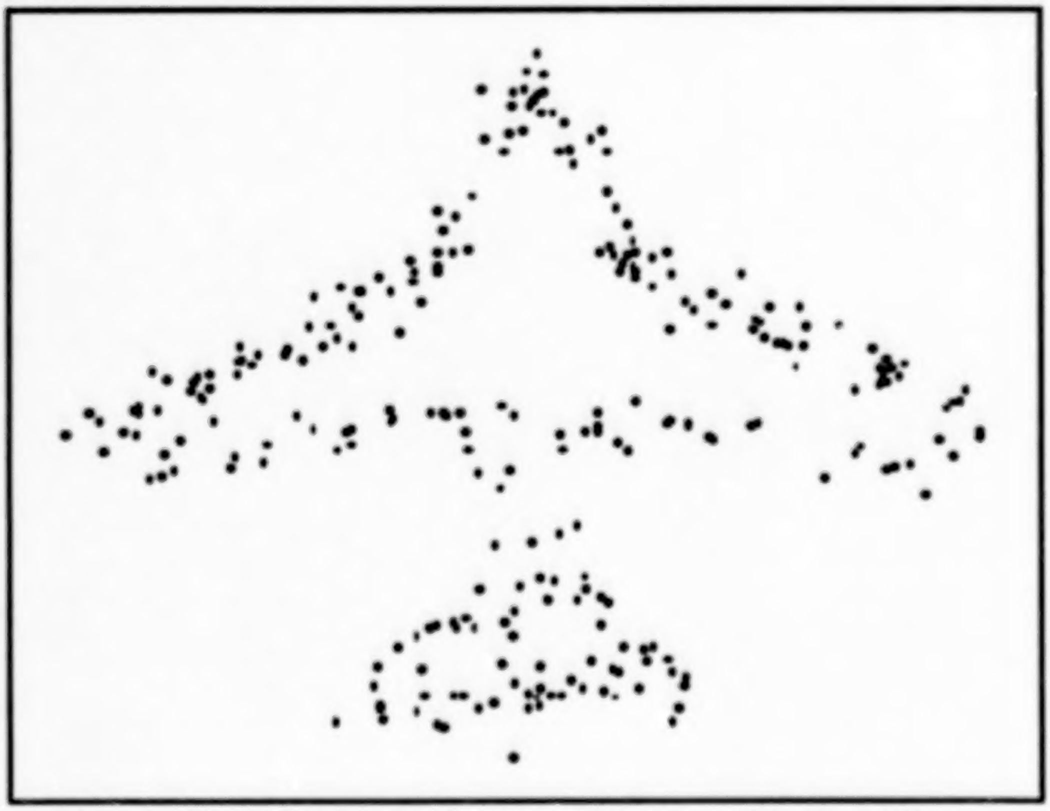


Figure 8(a) Aircraft 2 line object (10 db S/N)

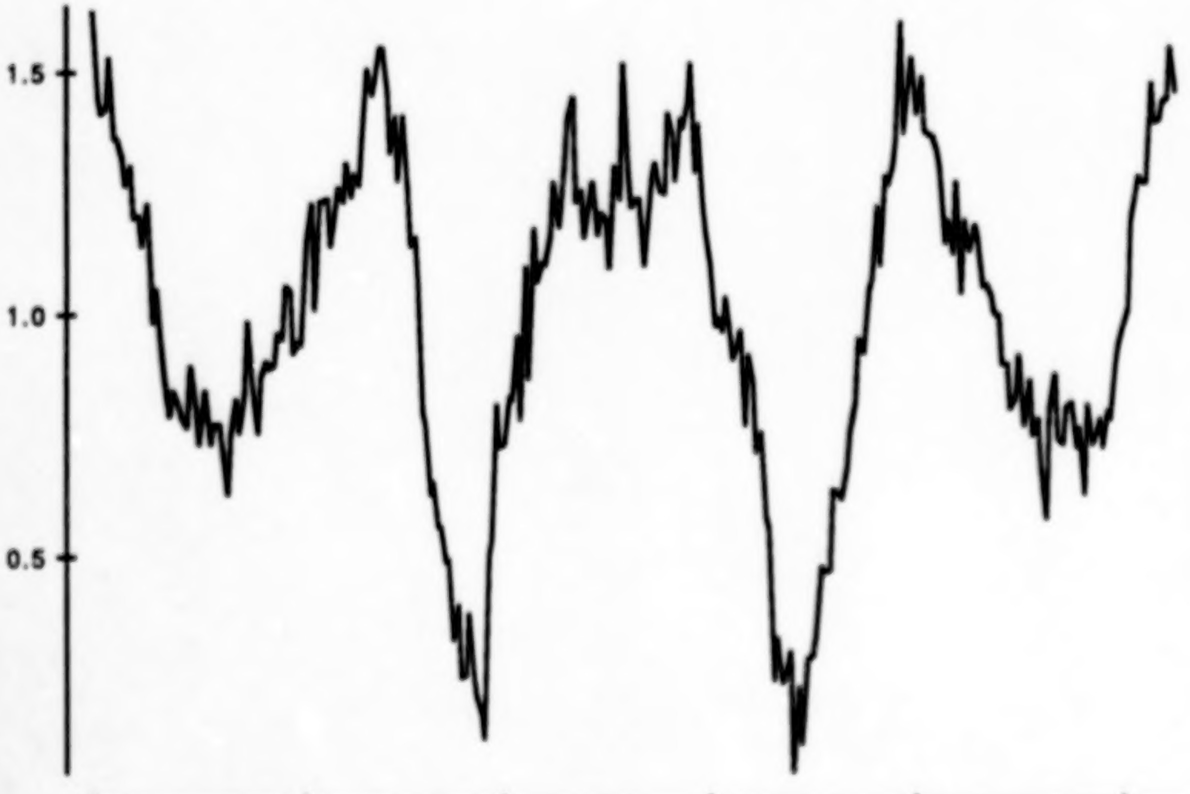


Figure 8(b) Aircraft 2 shape function (10 db S/N)

KNOWLEDGE-BASED IMAGING-SENSOR FUSION SYSTEM†

George Westrom
Odetics, Inc.
1515 S. Manchester Ave.
Anaheim, CA

INTRODUCTION

This paper describes an imaging system which applies knowledge-based technology to supervise and control both sensor hardware and computation in the imaging system. It includes the development of an imaging system breadboard which brings together into one system work that we and others have pursued for LaRC for several years. (Refs 1,2,3). The goal is to combine Digital Signal Processing (DSP) with Knowledge-Based Processing and also include Neural Net processing.

The system is considered a smart camera. Imagine that there is a microgravity experiment on-board Space Station Freedom with a high frame rate, high resolution camera. All the data cannot possibly be acquired from a laboratory on Earth. In fact, only a small fraction of the data will be received. Again, imagine being responsible for some experiments on Mars with the Mars Rover: the data rate is a few kilobits per second for data from several sensors and instruments. Would it not be preferable to have a smart system which would have some human knowledge and yet follow some instructions and attempt to make the best use of the limited bandwidth for transmission?

This paper will describe the system concept, current status of the breadboard system and some recent experiments at the Mars-like Amboy Lava Fields in California.

SYSTEM OVERVIEW

The system architecture concept is shown in Figure 1. The four areas shown are sensors, focal plane processor, knowledge-based supervisor/controller and image processors. Inputs to the system are supervisory commands, channel capacity and other mission data. The output is edited, classified and coded data, as well as other feature and range information.

Internal communications are provided so that sensors and processing can be supervised and controlled by the knowledge-based system. Rules, knowledge, data and researcher's expertise are contained in the Knowledge-Based Supervisor/Controller KBSC.

---

†This work was supported by the following NASA contracts: NAS1-18816 and NAS1-18664.

The data from multiple sensors are simultaneously processed and combined by the KBSC.

Signal and image processing will be performed both in the focal plane processor and with the image processor.

The parallel asynchronous processor and the KBSC and image processing hardware are described in other papers (Refs. 2, 3). The laser ranger system and goals of the KBSC system will be discussed here.

Figure 2 is a different view of the knowledge-based image/sensor fusion breadboard in that it segments the knowledge-based and signal processing in the manner in which it is implemented in the breadboard system. The knowledge base is hosted on a SUN computer and the real-time signal/video processing is on the Datacube pipeline image processor. The sensors are CCD cameras, IR sensors and laser ranger.

INTEGRATED LASER CAMERA

The integrated laser camera combines range data from the laser with the spatial reflectance data from the CCD camera. The concept is to provide a sensor which has both high spatial resolution and accurate range to specific objects in the scene. Table 1 summarizes the limitations of either sensor separately and the potential of fusing the data from each to give both range and spatial detail.

Table 2 lists the function of the ILC. The range may be provided at a single point or a range image may be generated by scanning the ranger over an area. Several display functions are available such as a contour map and an artificial grid to provide the concept of depth and range to any object in the scene. Camera control functions such as focus and zoom can be performed from the range output. Combining the range with high frequency spatial data can achieve rapid and very reliable camera focus.

Combining range and spatial data has many applications in image analysis, such as segmentation, distinguishing objects from shadows, obstacle avoidance, etc.

LASER RANGER

There are basically two types of lasers, pulse and continuous wave. Pulse laser rangers operate on the basis of measuring the time it takes a laser pulse to travel to the object and back to the receiver. A continuous wave (CW) laser ranger normally compares the phase shift between the transmitted and received wave. A discussion of the merits of CW vs. pulse is beyond the scope of this paper except to say that pulse laser rangers require less computation for range and generally are better for long-range imaging, for example, beyond 200 meters. Range ambiguity also needs to be resolved when using a CW laser ranger.

We selected a pulse laser shown in Figure 3 with the specifications shown in Table 3. Range, accuracy and firing rate were all important in selecting the laser ranger. The 2000 firings per second makes it quite feasible to generate small range images. An accuracy of about 5cm to 10cm can be achieved by averaging over a number of firings on a single object.

Figure 4 shows the block diagram implementation of the integrated laser/CCD prototype system. The laser ranger is mounted on a precision pan/tilt platform. The platform is controlled by the SUN either to point to a specific object or to scan an area to generate a range image. The camera focus and zoom will be controlled by the laser ranger in the breadboard imaging system. The video rate Datacube processor generates a graphics overlay on the video image such as the laser beam location.

Figure 5 shows the platform design for the ILC. The azimuth motion is provided with a rotary stage platform on which is mounted a goniometer for the elevation motion. The platform has a repeatability of about 0.17mR about each axis. The laser ranger will be bore sighted with the camera axis.

KNOWLEDGE-BASED SUPERVISOR/CONTROLLER

The objective of the KBSC is to provide a robust, flexible monitoring and control system which provides sensor control, processing and image coding control, outputs edited, classified and coded data based on an internal knowledge-base and data base, sensor input, and supervisory command.

Figure 6 shows some of the input, outputs and control functions of the KBSC system. The inputs to the KBSC can be from image processors such as the spatial frequency, histogram or other computed characteristics of the image. It may be a previously identified object or area so as to designate a small region of interest (ROI). Edge information and segmentation may be used to identify specific features in the image. The color or more generally the spectral response of the image may be used to identify regions or objects.

From the laser ranger, range and reflectance data may be used with the spatial data to identify features. Laser reflectance values may be used to determine the reliability of the range data and to determine the reflectance of the target at the laser frequency. The field of view (FOV) may be important when selecting processing algorithms. Sun angle, available bandwidth, and other priorities will be used to select processing algorithms and image coding methods.

In addition to sensor control such as focus, the KBSC will also be useful in applying the user's knowledge to select the information and data which is important.

AMBOY DATA COLLECTION

Some of the features of the breadboard imaging system are implemented in the Odetics Mobile Imaging Laboratory. Figures 7, 8, and 9 show the Mobile Imaging Laboratory platform with infrared and CCD cameras and platform with the laser ranger. Figures 10 thru 16 are samples of a set of images taken with the Mobile Imaging Lab at the Amboy Lava Field in the Mohave Desert which looks very much like Viking pictures of Mars. Figures 10 and 11 are CCD and laser scan images respectively of approximately the same area. Some of the very black lava in the lower right band corner did not reflect enough signal for the range computation. The bush in the center of the scene is clearly visible. (The black and white print was made from a pseudo color range image (Figure 11), causing the ranges at 50-60 meters to appear closer.)

Figures 12 and 13 are photographs of Amboy terrain which looks remarkably like the Mars Viking pictures except for a few bushes.

Figures 14 and 15 are infrared images (8-12μm) of the Amboy crater taken when the temperature was about 110°F. The white areas in the IR image are hot.

Figure 16 shows a panoramic scene of the Amboy Lava Field taken with a series of images with a narrow field-of-view camera on the pan/tilt platform. Figure 17 is an artist conception of how a Mars Rover camera might assemble narrow field-of-view pictures into a panorama. The image was digitized from an actual Mars Viking photograph.

REFERENCES

1. Tom N. Cornsweet, "Image Processing by Intensity-Dependent Spread (IDS)", International Workshop on Visual Information Processing for Television and Telerobotics, NAS CP-3053,1989. (Page 133 of this compilation).

2. D.D. Coon and A.G.U. Perera, "Parallel Asynchronous Systems and Image Processing Algorithms", International Workshop on Visual Information Processing for Television and Telerobotics, NAS CP-3053 1989. (Page 191 of this compilation).

3. E.R. Kurrasch, "Applications of the IDS Model", International Workshop on Visual Information Processing for Television and Telerobotics, NAS CP-3053 1989. (Page 165 of this compilation).

TABLE 1

INTEGRATED LASER RANGER/CAMERA SYSTEM

Problem:

Existing robot vision systems do not provide both range and high resolution.

- **Extracting range from the image data of CCD cameras is extremely computer intensive and provides poor 3-D contour data.**
- **Laser ranging/scanning systems provide range and 3-D contour data, but image detail is poor.**

Solution:

Provide intelligent fusion of high resolution CCD camera data and laser ranging data.

Use an expert system for determining how the data from different sources will be used to extract scene data.

TABLE 2

ILC FUNCTIONS

- Range to any point in scene
- Range image of any area in scene
- Display (range image, contour map)
- Display depth grid
- Display range to any point
- Focus camera
- Combine range and reflectance data

TABLE 3

501 LASER RANGER SPECIFICATIONS

Range	10-500 m
Accuracy	.2m
Resolution	.1m
Beam divergence	2.5 mR
Measurement rate	1–2000/sec
Weight	3 Kg
Power	3A @ 12V

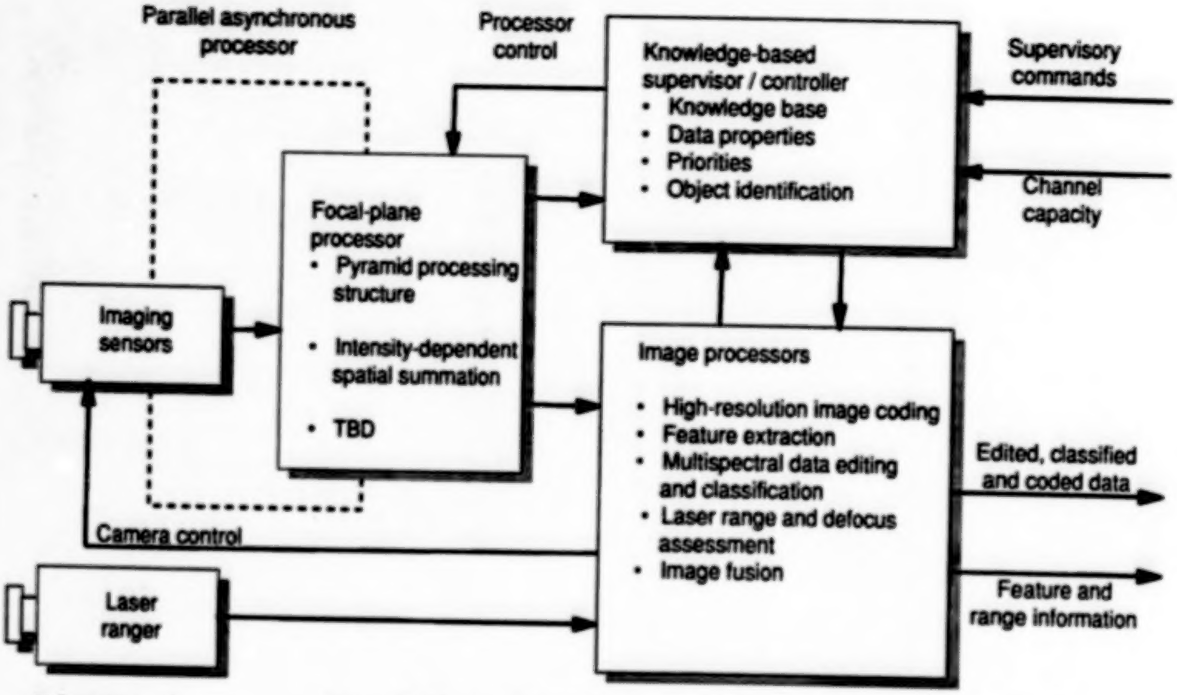

Parallel asynchronous processor
Processor control
Knowledge-based supervisor / controller
• Knowledge base
• Data properties
• Priorities
• Object identification
Supervisory commands
Channel capacity
Focal-plane processor
• Pyramid processing structure
• Intensity-dependent spatial summation
• TBD
Imaging sensors
Image processors
• High-resolution image coding
• Feature extraction
• Multispectral data editing and classification
• Laser range and defocus assessment
• Image fusion
Edited, classified and coded data
Camera control
Feature and range information
Laser ranger

Figure 1 Knowledge-Based Image/Sensor Fusion System

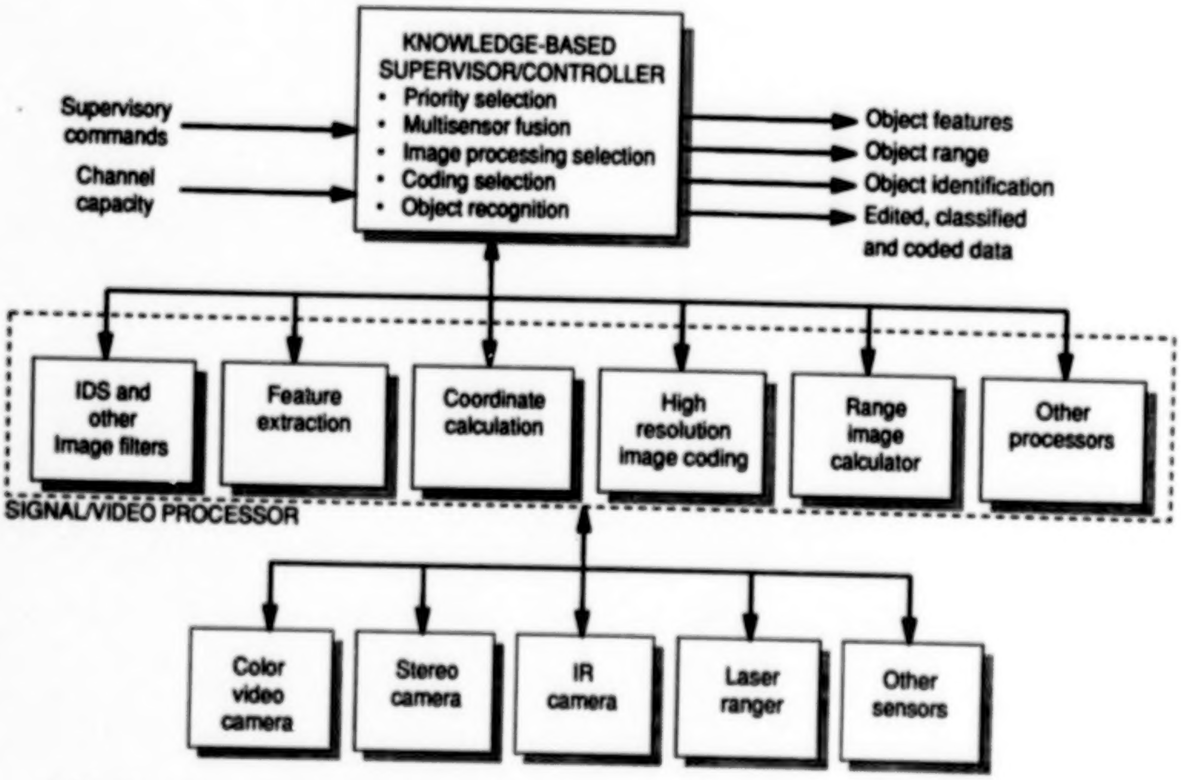

KNOWLEDGE-BASED SUPERVISOR/CONTROLLER
• Priority selection
• Multisensor fusion
• Image processing selection
• Coding selection
• Object recognition
Supervisory commands
Channel capacity
Object features
Object range
Object identification
Edited, classified and coded data
IDS and other Image filters
Feature extraction
Coordinate calculation
High resolution image coding
Range image calculator
Other processors
SIGNAL/VIDEO PROCESSOR
Color video camera
Stereo camera
IR camera
Laser ranger
Other sensors

Figure 2 Knowledge-Based Imaging/Sensor Fusion System

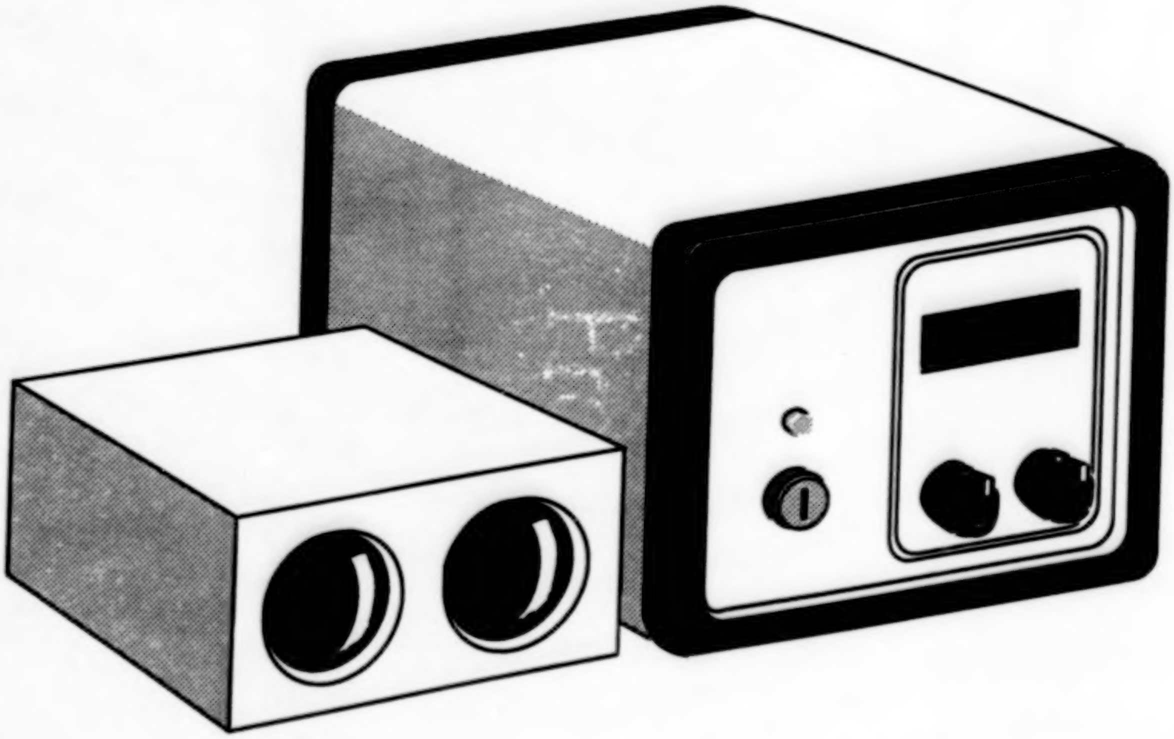


Figure 3 Rangefinder 501/SX

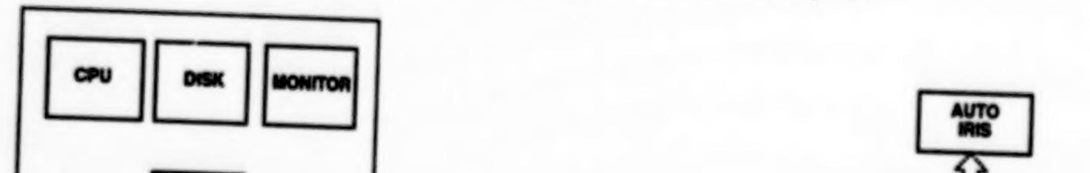

CPU
DISK
MONITOR
SUN
I/F
DATACUBE
I/F
PROCESSOR
MAXGRAPH
FRAMESTORE
DIGIMAX
COLOR MONITOR
AUTO IRIS
VID
VIDEO
ZOOM CONTROL
FOCUS CONTROL
LASER RANGER
PLATFORM
RANGE
TRIGGER
AZ COMMAND
AZ FEEDBACK
EL COMMAND
EL FEEDBACK
I/O BOARDS

Figure 4 ILC System Block Diagram

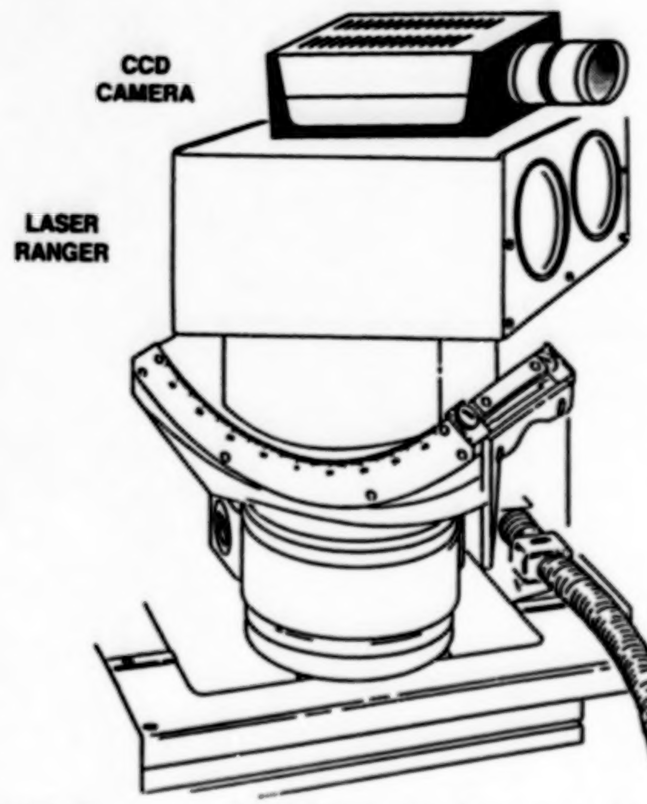

CCD
CAMERA
LASER
RANGER

Figure 5 Platform Design for the ILC

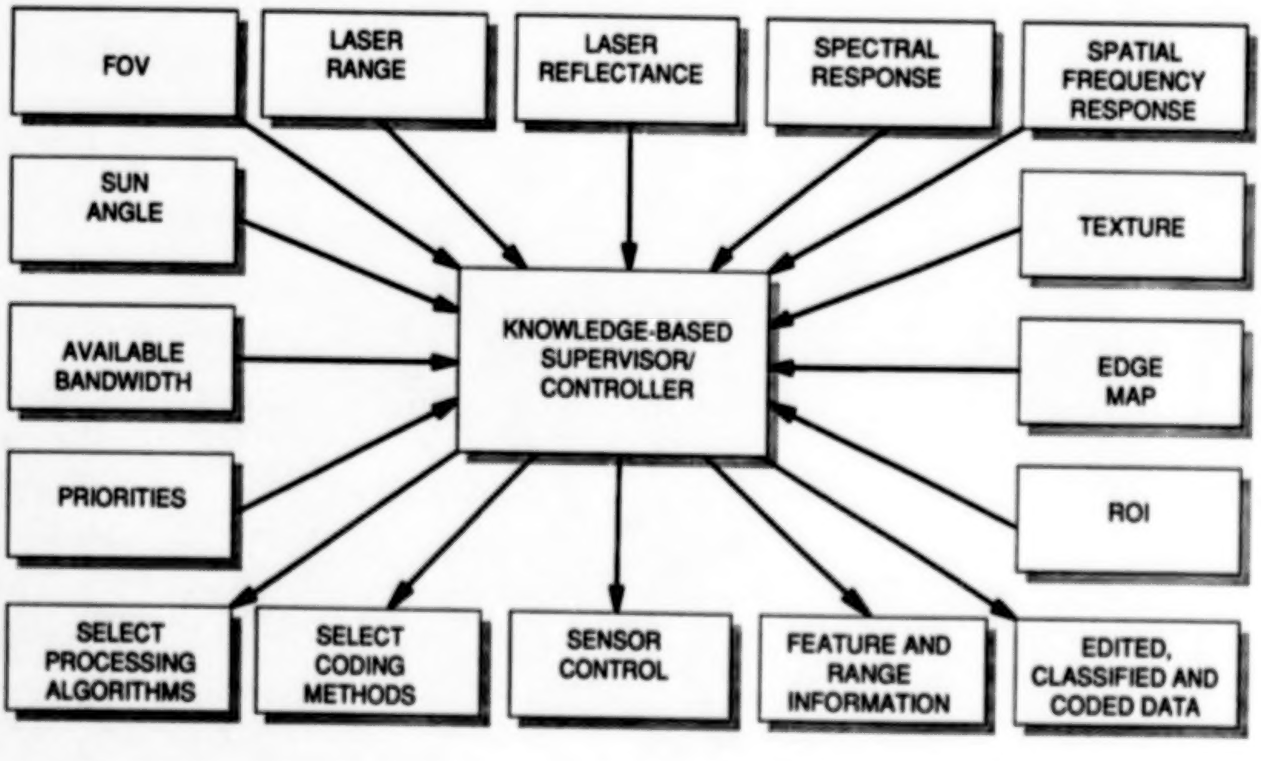

FOV
LASER RANGE
LASER REFLECTANCE
SPECTRAL RESPONSE
SPATIAL FREQUENCY RESPONSE
SUN ANGLE
TEXTURE
AVAILABLE BANDWIDTH
KNOWLEDGE-BASED SUPERVISOR/ CONTROLLER
EDGE MAP
PRIORITIES
ROI
SELECT PROCESSING ALGORITHMS
SELECT CODING METHODS
SENSOR CONTROL
FEATURE AND RANGE INFORMATION
EDITED, CLASSIFIED AND CODED DATA

Figure 6 Knowledge-Based Supervisor/Controller

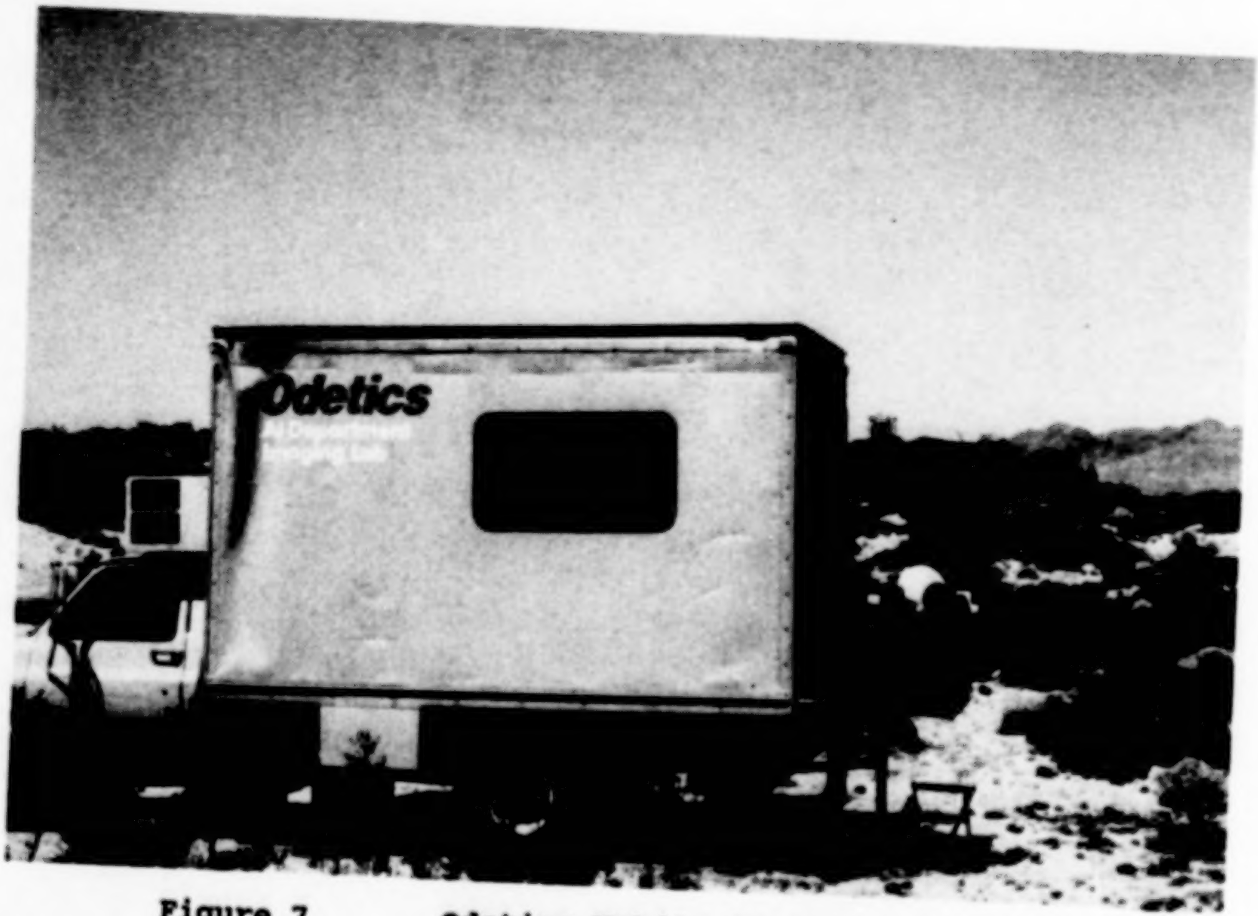


Figure 7 Odetics Mobile Imaging Laboratory

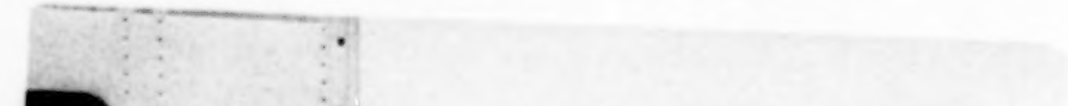


Figure 8 Sensor Platform with Infrared Camera

Figure 9 Sensor Platform with Laser Ranger

Figure 10 CCD Image Amboy Lava Field

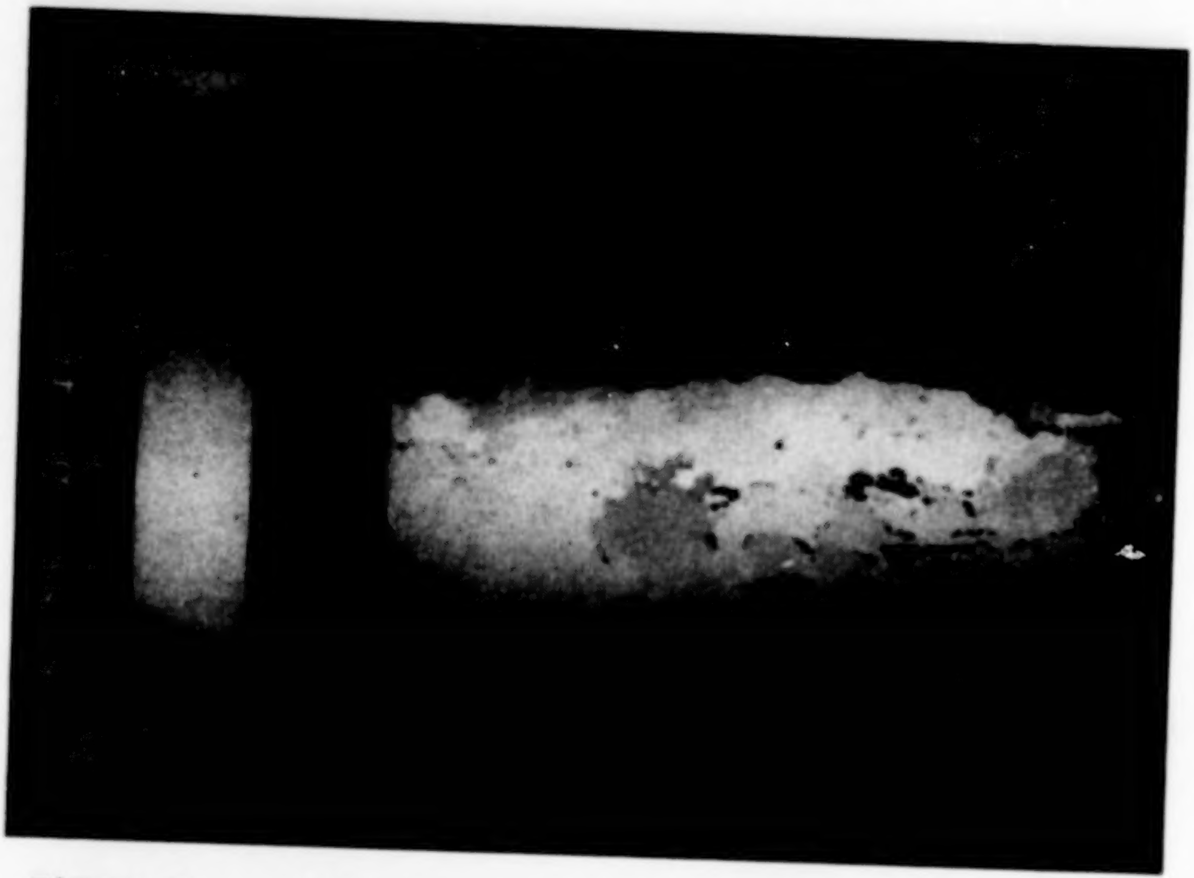


Figure 11 Laser Ranger Image of CCD Image (Figure 10)

Figure 12 Photograph of Mars-Like Amboy Lava Field

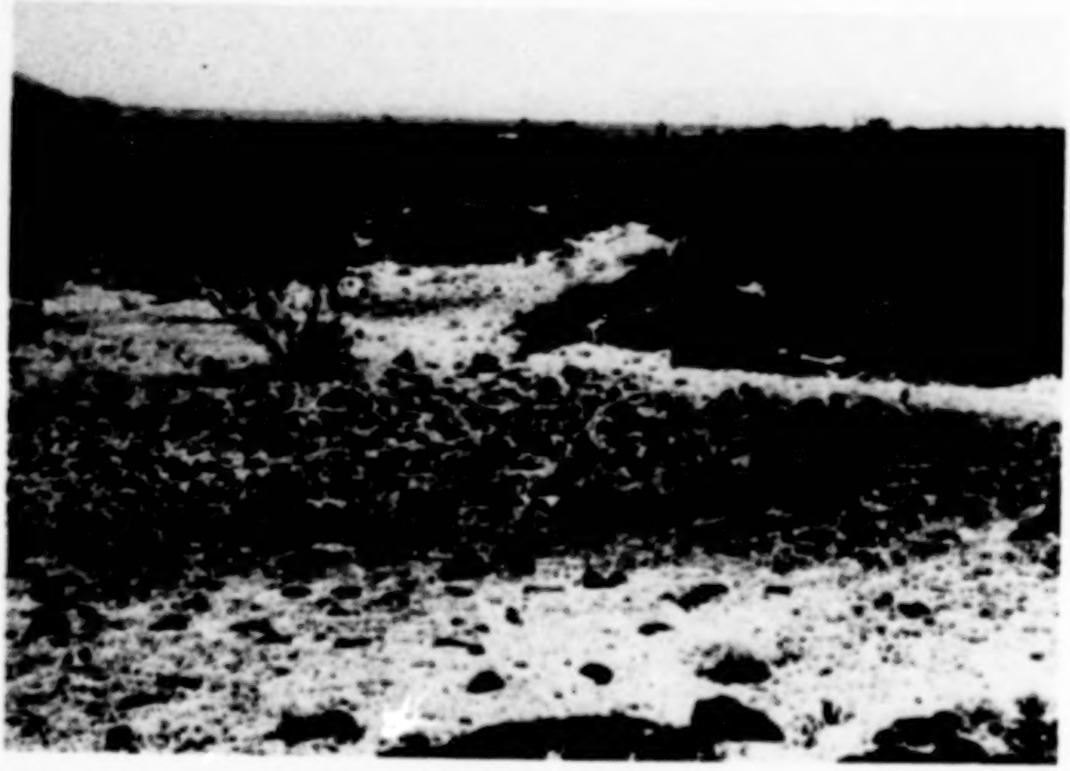


Figure 13 Photograph of Mars-Like Amboy Lava Field

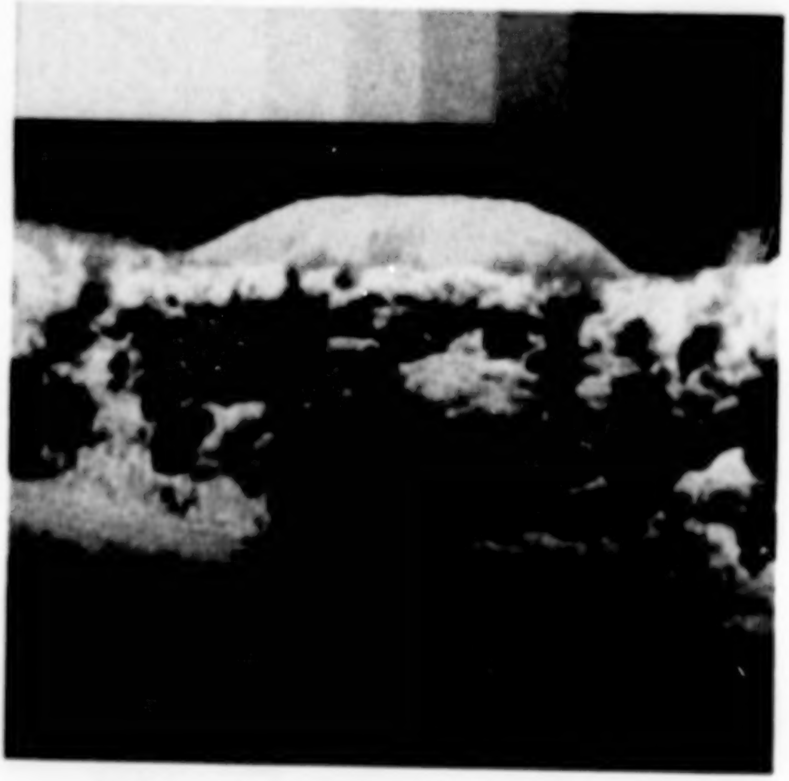


Figure 14 Infrared Image of Amboy Lava Field and Crater

Figure 15 Photograph of the same areas as Infrared Image, (Figure 14).

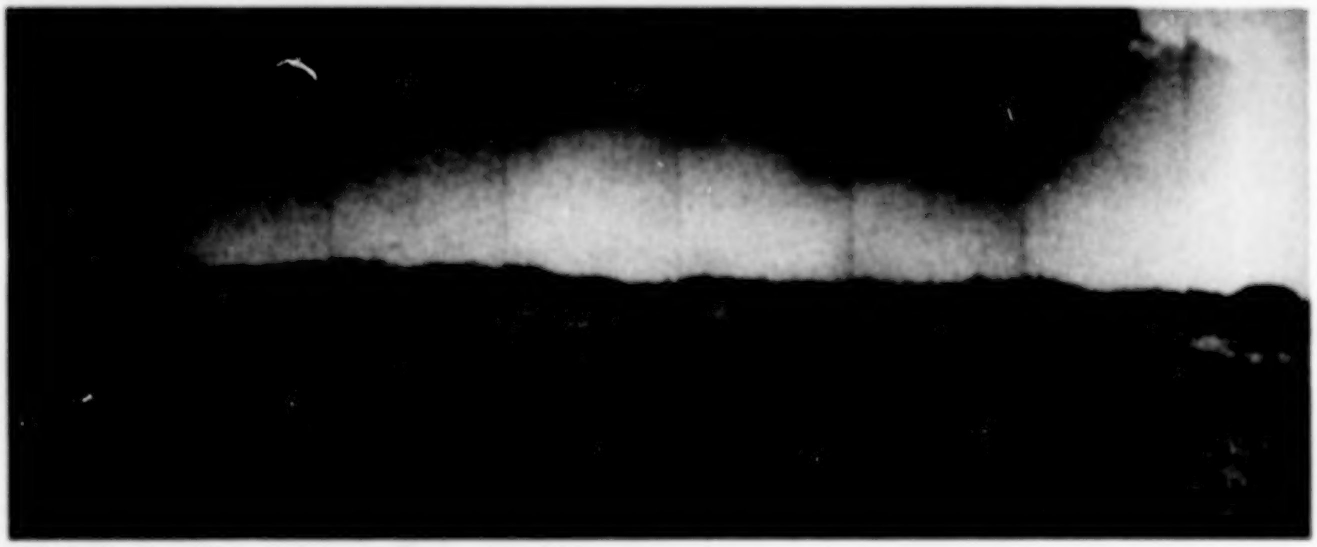


Figure 16 Panorama of Narrow Field-of-View Image of Amboy Lava Field

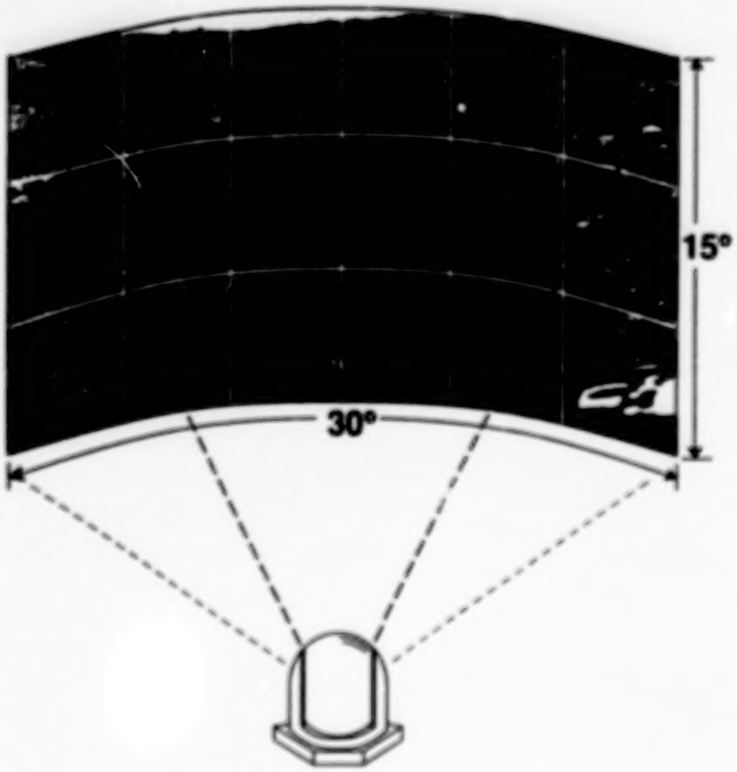


Figure 17 Artist's Conception of Mars Panorama Using Actual Viking Mars Image.

A KNOWLEDGE-BASED MACHINE VISION SYSTEM FOR SPACE STATION AUTOMATION[1]

Laure J. Chipman and H. S. Ranganath
Computer Science Department
University of Alabama in Huntsville

INTRODUCTION

A simple knowledge-based approach to the recognition of objects in man-made scenes is being developed. Specifically, the system under development is a proposed enhancement to a robot arm for use in the space station laboratory module. The system will take a request from a user to find a specific object, and locate that object by using its camera input and information from a knowledge base describing the scene layout and attributes of the object types included in the scene.

In order to use realistic test images in developing the system, we are using photographs of actual NASA simulator panels, which provide similar types of scenes to those expected in the space station environment. Figure 1 shows one of these photographs.

In traditional approaches to image analysis, the image is transformed step by step into a symbolic representation of the scene. Often the first steps of the transformation are done without any reference to knowledge of the scene or objects. Segmentation of an image into regions generally produces a counterintuitive result in which regions do not correspond to objects in the image. After segmentation, a merging procedure attempts to group regions into meaningful units that will more nearly correspond to objects.

Rather than taking this approach, we avoid segmenting the image as a whole, and instead use a knowledge-directed approach to locate objects expected in the scene. Constraints on the spatial relationships among objects and on attribute measurements of object types are used in obtaining a matching between regions of the input image and object descriptions in the knowledge base.

Section 2 describes the knowledge-based approach to scene analysis. Section 3 discusses the categories of knowledge used in our system. The remainder of the paper is a step by step description of the system under development.

KNOWLEDGE-BASED APPROACH

The use of a knowledge-based approach to object recognition is a growing area of research in image analysis. Use of knowledge improves recognition accuracy. We seek to avoid embedding this knowledge in the code, in order to create a more flexible system.

---

[1] This research is being supported by NASA Contract NCC8-16.

Knowledge of objects is traditionally used in the later stages of image analysis to match regions of an image with known objects. We are exploring the use of knowledge at earlier stages of the processing to help guide the search for objects.

A goal of our work is to provide a flexible system for locating objects, which could be updated for new scenes by simply adding to the knowledge base. The knowledge of scenes and objects is stored explicitly, rather than being embedded in the system's code. Objects and scenes are described in a general way, so that the system will not be overly sensitive to changes in the camera position or illumination. It is desirable to avoid exact models of the objects of interest. Some of the objects on the panels may be difficult to describe with a precise geometrical model. For example, the panel in Figure 1 contains switches that are enclosed in protective brackets. Because of their complicated structure and the existence of shadows, objects such as these will show up in the gradient image as a tangle of lines, easy to recognize but difficult to model geometrically.

There are many systems designed to match regions of an image to descriptions of objects stored in a knowledge base. McKeown's SPAM (System for Photo interpretation of Airports using MAPS) is one example [1]. This system takes the result of a traditional region-growing segmentation and attempts to group segments into meaningful objects. Levine and Shaheen describe a system in which segmentation is based on color, and regions are merged to form objects based on a long list of constraints on attribute measures of different object regions [2].

CATEGORIES OF KNOWLEDGE

For our application, the following categories of knowledge are used:

1) Knowledge of primitive, scene-identifying features
2) Measurement ranges of attributes of object types
3) Knowledge of spatial layout of scenes

In the first category, information about features consists of a list of procedures to be used to find the features, and parameters for these procedures. The scenes are described as lists of features that are present and absent from them. The information in this category was obtained through experimentation with input images. There is a need to develop an automated method for finding discriminating features for any new scene presented to the system.

The second category of knowledge consists of object types and ranges of acceptable values for attributes of those object types. The attributes used will preferably be invariant to scale or illumination changes and relatively insensitive to rotation. Such attributes as a texture measure, circularity, rectangularity, or ratio of length to width are good possibilities. A significant, but manageable, programming project would be to automate the gathering of these object attribute ranges, using a teacher to draw windows around several objects of a given type, and having the system automatically make and record measurements.

The third knowledge category contains information that aids in finding the starting points of probable objects. It contains the layouts of regions of the scenes to which input images will be matched. The knowledge in this category can help resolve ambiguities in the classification of objects by using spatial constraints. In other systems, this category could be expanded to include other types of constraints on the relationships among objects, such as adjacency or inclusion.

STRATEGY

The steps of the processing in our system are shown in Figure 2. The user indicates an object of interest. The system first verifies that the input image is a view that should contain that object. In the knowledge base, inclusion lists specify what panel contains each of the possible objects of interest. From this, we can determine which panel should be present in the input image. There is also a list of primitive features and parameters for procedures to find them listed in the knowledge base. The second step is then to identify which scene, or panel, is represented by the input image. The image is searched for distinguishing features to determine which scene is present. If the input image contains the scene of interest, we proceed to locate the object of interest. If not, the camera would be relocated to find the desired scene.

Once the proper scene is present, we find a region of interest within the scene. This region will contain the object of interest. The region of interest can be found relative to the location of the features found in the image in the scene identification step. Once the region is found, the camera can be made to zoom in on this area.

Within the region of interest, probable starting points to locate objects are found. Then, the boundaries of probable objects are found by searching windows around these starting points. Attributes of these probable objects will be measured. The knowledge base will list attributes of the different object types that are easy to recognize and identify. For each probable object in the region of interest, we obtain a list of object types for which the attribute measures match. Then, a matching between the input image and the scene layout described in the knowledge base must be found.

Although the actual procedures used for finding seed points, measuring attributes, and matching objects are specific to our application, these three steps could provide a useful starting point for other applications. For other image types, there could be other procedures developed for performing essentially the same three steps.

Preprocessing and Scene Identification

The first step in our processing is to obtain an edge image using the Sobel edge operators. This is done to facilitate locating boundaries of objects.

In our application, it is not likely that any significant rotation of the image will occur, since the camera will be mounted on a robot arm attached to a rail that runs the length of the module. Since the robot can know which end is "up," rotation is not a problem. In other cases, a system may need to deal with this possibility. For images of man-made objects such as control panels, a possible approach is to search Hough transform space for lines of maximum intensity. In scenes of control panels these are generally horizontal and vertical lines. Knowledge of the expected scene could also be used to determine at what angle the lines of maximum intensity should appear in the input image. This can be used to rotation-normalize the image.

Next, we identify which scene is present. We are assuming that an input image will contain one of a number of separate scenes. If the image contains parts of more than one scene, the process will generally not produce useful results. This goes along with the assumption that a camera attached to a robot arm could be positioned at a number of discrete, although approximate, positions along the length of the space station lab module.

To identify the scene, the system searches for primitive distinguishing features. The presence or absence of the features in the input image is matched with lists of features present for each scene in the knowledge base. Presently, a scene must have features that match exactly with one of the scenes in the knowledge base. It would be possible to allow for closest matches by computing the string distance between a binary string denoting presence and absence of features in the input image with the strings in the knowledge base, and assume the scene to be the one with the closest match.

The features used for scene identification are sets of lines in the gradient image with certain characteristics. These lines correspond to edges in the original image. A line in our system is defined in terms of a merit measure which is a linear combination of average intensity and average difference between successive pixels along the line. A row of pixels of high intensity and low average difference is a "good" line. The characteristics of intensity and average difference can be useful taken separately. The average difference measure provides a good measure of texture which is easy to compute. Some of the features used for scene identification are lines of high average difference.

Figure 3 depicts the scene identification process for the panel of Figure 1. In this example, lines of high texture, as measured by high average difference, are found through the columns of an array of lights, and also through a row of switches. The diagonal line and the set of lines in the upper left corner of the image represent the best matches for two additional features that are present on other panels but not on this panel.

We use primitive features to keep processing for scene identification to a minimum, but any features could be used, as long as the process for finding them could be listed in the knowledge base.

Object Seed Points

Given the location of features in an input image, it is possible to compute coordinates for a region of interest of the scene that contains the desired object.

Once an image of the region of interest of the scene is obtained, we find starting points of probable objects. In scenes consisting of well-separated blobs on a background, a method that has proven useful is to search for a specified number of horizontal and vertical lines of high texture, with some minimum spacing between them. Figure 4 shows the result of this process on one of our regions of interest. Most of the intersection points pass through objects on the image. There are some false lines, since there is some printing on the control panels that results in high-texture lines.

The minimum spacing chosen is large enough to prevent the appearance of more than one line in the same direction through the same objects. Only a minimum is given so that the object seed points may be found for images that are translated or scaled differently.

The intersection points of lines found are possible object locations. For other types of images, other methods for finding seed points of objects could be used. If an object's color is known, the image could be searched to find a patch of that color as a starting point for a region-growing routine. Likewise, any other attribute of an object, such as intensity or texture could be used to find a patch from which to start a region-growing routine. This may be better than performing a global segmentation and growing all possible regions in the image, which probably do not correspond to objects.

We are experimenting with methods of finding object boundaries within windows centered on the seed points. The use of these seed points can reduce computation by limiting the search area for objects.

Measurement of Attributes

The control panels contain instances of a finite number of object types, e.g. switches, buttons, knobs, etc. For each object type, the knowledge base contains an acceptable range of values for each attribute. The attributes used may differ depending on the object type. For example, circularity may be a good attribute to use for knobs, but texture may be better for switches enclosed in brackets. Since the possible objects in the input image may be processed in parallel, it may be worthwhile to measure all attributes, even though some results may be not be used. Once the attribute measures have been determined, the knowledge base is consulted to determine for each possible object the set of object types consistent with its measurements. For example, Object 1 may "look like" a switch or a button. Some possible objects will not match to any object types.

The result of this process is a grid showing the possible objects which could be located at each point.

Matching Scene Layout

Our system will match the input image with a grid layout of the region of interest in the knowledge base. The points on the grid correspond to intersection points of lines passing through the objects. Some points will not correspond to any object, but pass through empty space.

Figure 5 shows examples of knowledge base and input grids. The input image will be processed to produce a grid layout of what is found. The matching routine will find a consistent match between the knowledge base grid and the input grid. In general, the input grid may have more rows or columns than the knowledge base grid. There may be non-object points in the input that happen to look like a certain object type based on their attribute measures. The constraint of the layout given in the knowledge base will help to find a consistent matching. In theory, there could be more than one consistent matching for a given scene, but the fact that both attribute measures and scene layout constraints are used will reduce the chance of an incorrect matching.

We are producing a deterministic matching routine, but this may be expanded to find a closest match, thus enabling the system to handle partially occluded objects or problems with glare.

The constraint of scene layout, meaning left-right, above-below relationships is not the only constraint that could be used to find a consistent match. There are other scene attributes that can be represented in graph form that constrain the interpretations of the scene. Adjacency and inclusion relationships are two examples.

There has been some work done to develop a theoretical basis for graph matching. Shapiro and Haralick [3,4] have developed a graph theoretic method of partial matching, using distances between graphical descriptions of input images and those on file for known images. They apply this approach to matching relational descriptions of objects with their descriptions stored in a knowledge base. The same idea can be applied to matching relational descriptions of scenes in which the objects are stationary. This is a promising approach for handling problems of missing or occluded objects or variations due to noise. More flexibility is provided by the matching of attributed graphs. Sanfelieu and Fu [5] have described a distance measure between attributed graphs which may be useful. Their work is applied to syntactic recognition of objects, but could also be applied to graphical descriptions of scenes.

CONCLUSION

A system that uses knowledge of scenes and objects to aid in segmentation and location of desired objects is being developed. The system is not based on any geometrical modeling of objects or on precise measurements of object location. A goal was to make the system relatively insensitive to changes in camera position and illumination, taking into account the fact that a robot's positioning system will not be perfect. The approach used is most applicable to scenes in which objects are stationary and well-spaced, such as control panels.

The usual approach to object recognition is to segment the entire image and then try to make sense out of all the segments by matching them to known objects. In our approach we eliminate needless processing of segments that do not correspond to known objects. We change the focus of attention of the system based on information about the scene layout, to match up only objects that will assist in finding the object of interest. Once we have a consistent mapping of areas of the image to known objects, we have completed processing. Other features in the image are ignored.

Although this system is designed specifically to process man-made scenes such as control panels, in which objects are usually well-separated on a background, the basic idea can be generalized to other applications. In any application in which the scenes consist of fixed objects or regions, knowledge of scene layout can be used to direct the segmentation process and to constrain possible interpretations of the objects found in the scenes. Different approaches can be found for determining object seed points, and then for growing regions from points identified as being likely parts of objects of interest. Different attributes of these regions can be measured for different applications. Constraints on relationships among objects other than the simple spatial layout can also be used.

REFERENCES

[1] D. McKeown, Jr., W.A. Harvey, Jr., and J. McDermott, "Rule-based Interpretation of aerial imagery," IEEE Trans. Pattern Analysis and Machine Intelligence, PAMI-7, No. 5, 1985, pp. 570-585.

[2] M. Levine and S. Shaheen, "A modular computer vision system for picture segmentation," IEEE Trans. Pattern Analysis and Machine Intelligence, PAMI-3, No. 5, 1981, pp. 540-556.

[3] L.G. Shapiro, R.M. Haralick, "A metric for comparing relational descriptions," IEEE Trans. Pattern Analysis and Machine Intelligence, PAMI-7, No. 1, 1985, pp. 90-94.

[4] L.G. Shapiro, R.M. Haralick, "Structural descriptions and inexact matching," IEEE Trans. Pattern Analysis and Machine Intelligence, PAMI-3, No. 5, 1981, pp. 504-519.

[5] A. Sanfelieu, and K.S. Fu, "A distance measure between attributed relational graphs for pattern recognition," IEEE Trans. Systems Man Cybernetics, SMC-13, No. 3, 1983, pp. 353-362.

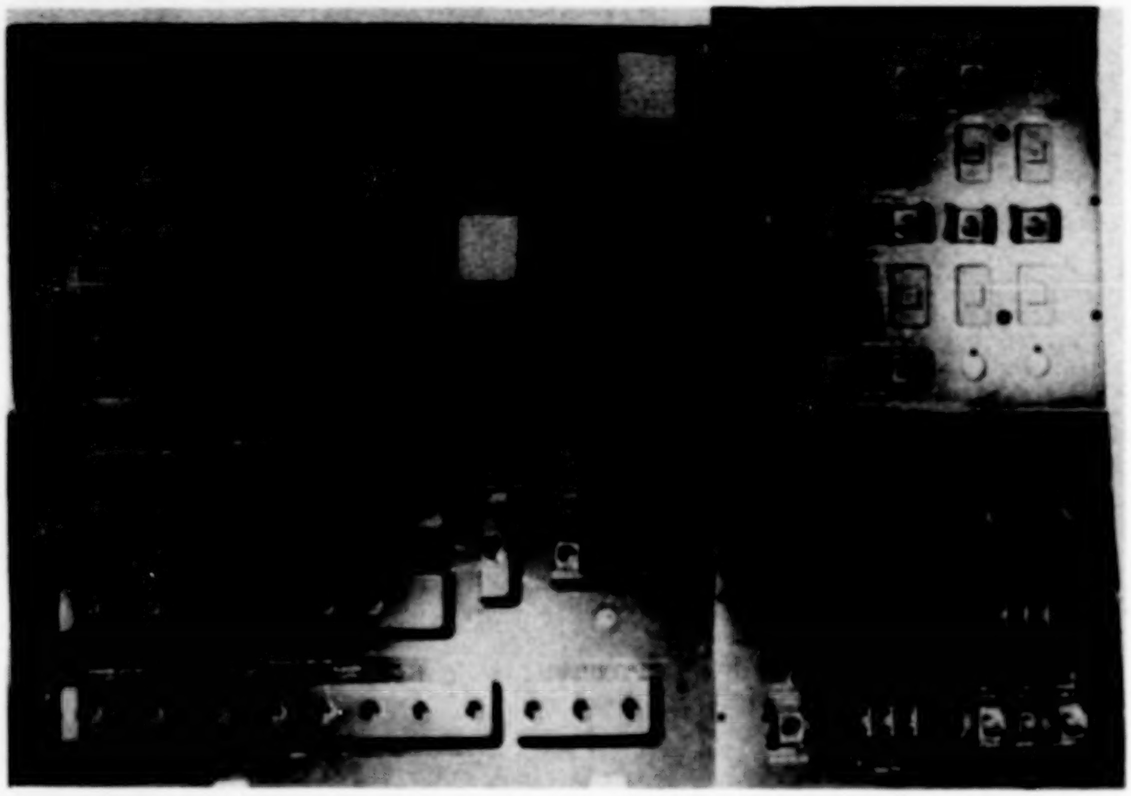


Figure 1: One of the scenes used as a realistic test image for the system.

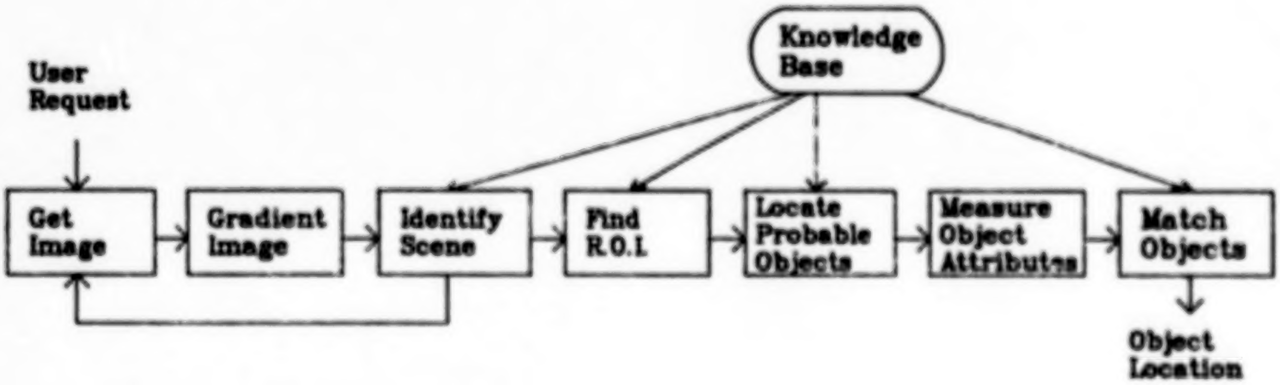

Knowledge Base
User Request
Get Image
Gradient Image
Identify Scene
Find R.O.I.
Locate Probable Objects
Measure Object Attributes
Match Objects
Object Location

Figure 2: The steps in the object recognition process.

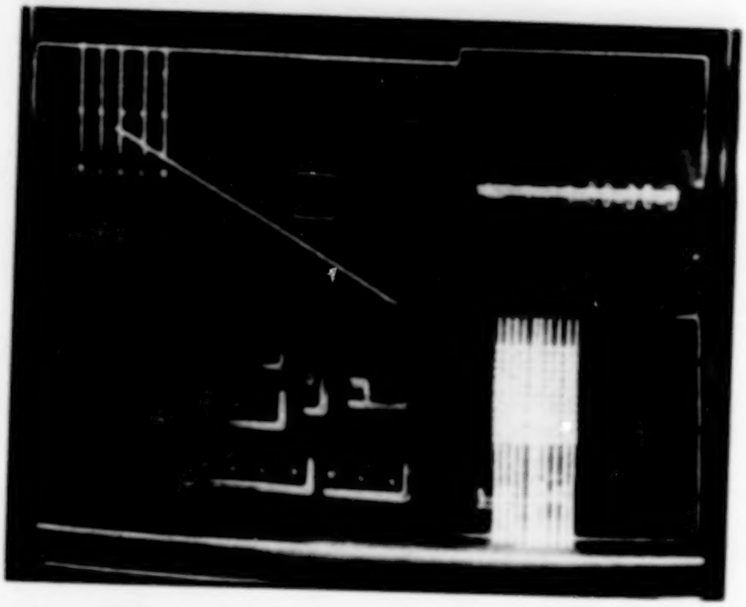


Figure 3: The scene identification process performed on the gradient image of the panel in Figure 1.

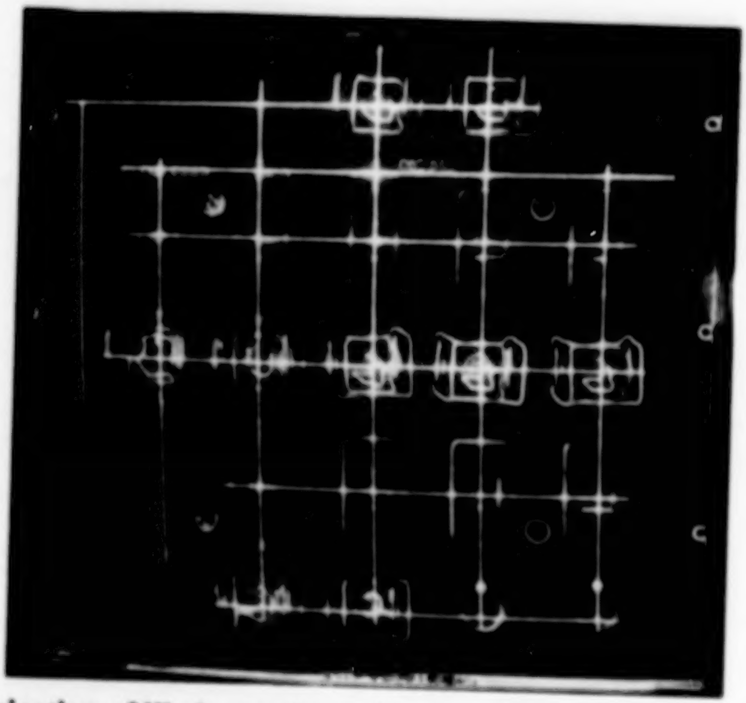


Figure 4: Determination of likely starting points for objects, performed on a sub-panel of Figure 1.

INPUT GRID

–	–	Switch, Knob
Button	–	–
Rectangle	Rectangle	Rectangle
Switch, Button	Switch	Switch, Button

KNOWLEDGE BASE GRID

–	–	Switch 1
Rectangle 1	Rectangle 2	Rectangle 3
Switch 2	Switch 3	Switch 4

Figure 5: A hypothetical grid representing possible object layout in an input image, and a grid from the knowledge base to be matched to it.

A Programmable Image Compression System

Paul M. Farrelle

Optivision, Inc.
2655 Portage Bay Ave., Davis, CA

Abstract

This paper describes a programmable image compression system which has the necessary flexibility to address diverse imaging needs. It can compress and expand single frame video images (monochrome or color) as well as documents and graphics (black and white or color) for archival or transmission applications. Through software control the compression mode can be set for lossless or controlled quality coding; the image size and bit depth can be varied; and the image source and destination devices can be readily changed. Despite the large combination of image data types, image sources, and algorithms, the system provides a simple consistent interface to the programmer. This system (OPTIPAC™) is based on the TI TMS320C25 digital signal processing (DSP) chip and has been implemented as a co-processor board for an IBM PC-AT compatible computer. The underlying philosophy however can readily be applied to different hardware platforms; and by using multiple DSP chips or incorporating algorithm specific chips the compression and expansion times can be significantly reduced to meet performance requirements.

Introduction

The goal of image data compression is to squeeze out the redundancy in a digitized image (B bits/pixel) such that the compressed image can be represented by $b < B$ bits/pixel, but can later be expanded to give an output image containing B bits/pixel. The ratio B/b is called the compression ratio (CR) and the challenge is to maximize the CR given a set of constraints. Typical constraints are: lossless fully reversible compression so that the output and input images are bit for bit identical; a fixed maximum execution time; a minimum subjective output quality level.

Image data compression has become an integral part of imaging systems despite improvements in transmission and storage capacities. This is because imaging systems continue to use higher resolution sensors and the size of image data bases is growing rapidly. In some cases it is economically attractive to incorporate compression into an imaging system, while other times its use is mandatory to achieve the overall system specifications.

In this paper we describe a compression and expansion system which is rapid but does not operate at real-time rates. We have previously reported a compression system suitable for real-time TV and videoconferencing applications [1]. The current system processes each frame without reference to any other frame and is therefore an intraframe, or single frame, coder. Single frame imaging applications include: telemetry, remote surveillance, image databases, picture ID systems, medical imaging, law enforcement, electronic publishing, digital mapping, insurance claims, parts catalogs, point of sale systems, etc. [2]. Each application has different requirements and even within a single application there is often the need to handle multiple image data types, sources, and compression modes. Image data types might include monochrome with 8, 10, or 12 bits per pixel; color with 16 or 24 bits per pixel; and binary images which can only take one of two values. Image sources are also diverse and images could come from computer files, frame grabbers which can capture a single video frame, image scanners, and communication links. Finally the application might require a lossless compression mode, or this constraint might be relaxed to allow small differences between the output and original images to increase the CR. In this controlled quality case there is a trade-off to be made between the output image quality and the CR. In either case the mode can be further specified as being high speed or high compression ratio. In the former case the algorithm is kept simple so as to achieve a high speed compression or expansion. In the second case the emphasis is to achieve the highest CR possible and as expected this will take longer.

It is clearly desirable to have a single system which can handle multiple image data types (binary, gray scale, color) at multiple resolutions and provide different compression modes (lossless, controlled quality). The overall system requirement is then to provide all of this flexibility via a consistent interface.

OPTIPAC™ Hardware Specifications

A system has been built to provide this capability in a PC environment. It is a co-processor board for an IBM PC/AT compatible computer and is built around the TMS 320C25 DSP chip operating at a 40MHz clock speed. Local memory consists of 32 or 64 kbytes of high speed static RAM with no wait states. The memory is used to store the current compression application and provide a data area where the current image window can be processed. The memory is dual ported so that it can be accessed by both the DSP and the AT host. The AT downloads the appropriate compression or expansion application code at the beginning of a session and then compresses an image, a window at a time, by repeatedly loading image windows and unloading compressed data blocks. For more exacting requirements, multiple OPTIPAC™s can be operated in parallel, within a single system, to reduce the overall execution time.

Software Requirements

The major requirement was to provide a consistent interface which would be independent of the compression algorithm, the image data type, and the image source and destination device types. A second important requirement was to allow algorithms to be selected via descriptive terms such as controlled quality, level 6 rather than "optimal adaptive widget coding with a threshold of 32.452 followed by customized variable length coding". This has the obvious advantage of not requiring a user to be a coding expert, but it also provides one level of indirection so that different algorithms can be substituted at a later time without changing high level application code.

Indirect Algorithm Selection Table (IAST)

To implement the descriptive approach to algorithm selection, we created an Indirect Algorithm Selection Table (IAST). This table translates descriptive terms plus the image data type into a 4-bit index which then maps into a specific algorithm. By combining this table with the DSP application code, algorithms can be changed or updated without recompiling any of the user's application code. All that is required is a new DSP application file which can readily be distributed to existing users. The current IAST is shown in Table 1 and the algorithms which have been implemented are characterized as follows [3]:

A10 - Lossless high compression ratio algorithm for monochrome or RGB color images. Based on predictive and variable length coding.

A20 - High speed lossless coding algorithm for monochrome or RGB color images. This is also based on predictive and variable length coding, but uses simpler versions to increase the speed.

A30 - High compression ratio controlled quality algorithm for monochrome images. Based on adaptive cosine transform coding.

A31 - Same as A30 but for RGB color images.

A40 - High speed lossless coding algorithm for binary images. Based on CCITT one-dimensional RL coding.

A50 - Lossless high compression ratio algorithm for binary images. Based on CCITT modified READ coding.

The Universal Algorithm Interface (UAI)

The OPTIPAC™ compression system is shown in Fig. 2. In this diagram the PC host memory is represented by the central block and contains a compression application which has been linked with the OPTIPAC™ run time library to form an executable image. The interface between the user's application and the run time library is depicted as the universal algorithm interface (UAI) and represents a set of function calls which are used to control a compression or expansion session. At the top left of the diagram we see image data stored in a file and also a frame buffer. The *display.cnf* file associated with the frame buffer is simply a configuration file needed by the system. At the top right we see the destination for the compressed data, a second disk file. This is simply one example configuration and different sources and destinations are of course possible. At the bottom of the diagram we see the compression hardware and a disk file which contains the DSP applications and the IAST.

Stage I - Setup The first stage of a compression session is the setup. This is accomplished by a call to the SetupCompress function using the following syntax:

SetupCompress(*x, y, z, nx, ny, nz, color, bits, type, mode, quality*)

This specifies to the compression system that we wish to process a region of size (*nx, ny, nz*) at offset (*x, y, z*) where *x* and *y* are spatial coordinates within the current frame, and *z* is the frame number. The image data type is specified by *color* which specifies color or monochrome and *bits* which is the number of bits per pixel. Then finally the *mode* and *quality* parameters specify the compression mode and one of the ten controlled quality coding levels: 9 (highest), 8, ..., 0 (lowest). This function call initializes the compression code and, as shown in Fig. 3, causes the appropriate DSP application (algorithm) to be downloaded to the compression engine.

Stage II - Compress The next step is to compress the image region specified in the setup stage. This is accomplished by a call to the Compress function using the following syntax:

Compress(*read_window, write_block*)

This provides the compression system with two user supplied I/O functions: *read_window* and *write_block* which are independent of the specific algorithm in use. In Fig. 4 we have:

Compress(*ReadFileWindow, WriteFileBlock*)

and *ReadFileWindow* is called by the run time library to load image data into the compression hardware and then, after processing, the compressed data is stored using the *WriteFileBlock* function. By simply changing the I/O functions, unlimited image sources and destinations can be accommodated. For example in Fig. 5 we have:

Compress(*ReadFrameWindow, WriteFileBlock*)

and the image source is now a frame buffer rather than a disk file.

Since the compression hardware is unable to store the complete image on board, the *read_window* and *write_block* functions are called repeatedly to process the complete image region specified in Setup. Furthermore the window size requested each time by OPTIPAC™ is algorithm dependent and is made as large as possible to minimize the overhead associated with a data transfer between the PC host and OPTIPAC™. The run time library code handles this complicated algorithm dependent control leaving a simple consistent interface, the UAI. The complete system is shown in Fig. 6.

Performance

Overall system performance is shown below in Tables 2 and 3. In Table 2, CR is shown when images with different data types and of differing complexity are compressed using some of the available modes. Five different compression modes are shown: **hs** - high speed lossless; **hc** - high compression ratio lossless; and q5, q3, q1 which are controlled quality coding at quality levels of 5, 3, and 1 respectively. The corresponding execution times are shown in Table 3.

Typical times for compressing 512x512 monochrome images on a standard 8 MHz IBM PC/AT are: 2 seconds for the **hs** mode and 6 to 7 seconds for the controlled quality mode. Coding 512x512 color images takes 5 to 6 seconds for the **hs** mode and 9 to 10 seconds for the controlled quality mode. Compressing binary 8.5"x11" pages sampled at 200 dots per inch and 200 lines per inch takes about 2 seconds for **hs** and 4 seconds for the **hc** mode. Expansion times are similar to compression times.

Note that although the CR figures for **hc** coding are always greater than, or at least equal to, the corresponding **hs** figures, they are often not significantly larger for monochrome and color images. In fact, **hc** only provides substantial improvements when the **hs** figures are already relatively large. Furthermore **hc** coding can take much longer than the other modes because it is currently implemented on the slow PC/AT host. If the application is long term archival and the only requirement is to obtain the highest possible CR then the **hc** option should always be used. Otherwise, only when the noise level is low and there is a great amount of redundancy in the image, that is the **hs** CR is 2 or more, is it usually worth spending the extra time to utilize the **hc** option. Consider the color baboon and logo images as extreme examples. The baboon takes an extra 72.5 seconds to compress in **hc** mode rather than **hs** mode, but still only achieves the same CR (=1.4). On the contrary, the computer generated logo image takes only an extra 10.5 seconds to dramatically increase the CR from 2.4 to 23.2.

Conclusions

We have described a programmable image compression system that can handle images of any type and size while maintaining a consistent interface. For systems requiring even faster performance, multiple boards can be used or the consistent design philosophy can be extended to more complex systems containing multiple processors and compression specific hardware modules.

References

1. Jain A. K., and D. G. Harrington, "A 10 MHz data compression system for real-time storage and transmission", Appl. of Digital Image Processing VIII, Proc. SPIE 575, pp. 62-65, 1985.

2. Farrelle, P. M., D. G. Harrington, and A. K. Jain, "Image data compression in a personal computer environment", Appl. of Digital Image Processing XI, Andrew G. Tescher, Ed., Proc. SPIE 974, pp. 177-186, Dec. 1988.

3. A. K. Jain, "Fundamentals of digital image processing",Chapter 11, Prentice Hall, 1989.

graphics	color	high compression	controlled quality	index	algorithm
0	0	0	0	0	A20
0	0	0	1	1	A30
0	0	1	0	2	A10
0	0	1	1	3	A30
0	1	0	0	4	A20
0	1	0	1	5	A31
0	1	1	0	6	A10
0	1	1	1	7	A31
1	0	0	0	8	A40
1	0	0	1	9	A30
1	0	1	0	10	A50
1	0	1	1	11	A30
1	1	0	0	12	A20
1	1	0	1	13	A31
1	1	1	0	14	A10
1	1	1	1	15	A31

Table 1 Indirect Algorithm Selection Table (IAST). The image data type (graphics, color) and compression mode (high compression, controlled quality) are used to form a 4-bit index into the IAST. The first bit is 1 for graphics images and 0 for non-graphics images; the second bit is 1 for color and 0 for monochrome; the third bit is 1 for high compression ratio mode and 0 for high speed mode; the fourth bit is 1 for controlled quality coding and 0 for lossless coding. The specific algorithms (A10, A20, A30, A31, A40, and A50) are described in the text.

	hs †	hc ‡	q5 *	q3 *	q1 *
Monochrome images (512x512x8)					
Very Simple Scene (adac)	2.8	3.9	32	45	115
Simple Scene (f18)	1.7	1.9	14	21	49
Complex Scene (airport)	1.3	1.5	8	12	31
Color images (512x512x16)					
Very Simple Scene (AT&T logo 512x400)	2.4	23.2	31	46	111
Simple Scene (lenna)	1.8	2.0	22	35	92
Complex Scene (baboon)	1.4	1.4	10	17	61
Black and White images (1728x2376x1)					
Simple Scene (CCITT 2)	12.5	18.8	-	-	-
Complex Scene (CCITT 7)	4.5	5.3	-	-	-

† hs - High Speed lossless compression mode

‡ hc - High Compression Ratio lossless compression mode

* q5, q3, q1 - Controlled Quality (Quality Levels: 5 (highest), 3, 1 (lowest))

Table 2 Compression ratios (CR) for different image data types and compression modes.

	hs †	hc ‡	q5 *	q3 *	q1 *
Monochrome images (512x512x8)					
Very Simple Scene (adac)	1.8	15.5	5.8	5.6	2.4
Simple Scene (f18)	2.2	29.8	7.6	6.7	3.0
Complex Scene (airport)	2.1	36.4	10.1	8.5	3.6
Color images (512x512x16)					
Very Simple Scene (AT&T logo 512x400)	4.5	15.0	7.5	7.1	3.5
Simple Scene (lenna)	5.9	63.5	10.4	9.0	3.6
Complex Scene (baboon)	5.9	82.4	15.5	12.0	4.1
Black and White images (1728x2376x1)					
Simple Scene (CCITT 2)	1.7	1.8	-	-	-
Complex Scene (CCITT 7)	3.9	4.6	-	-	-

† hs - High Speed lossless compression mode
‡ hc - High Compression Ratio lossless compression mode
* q5, q3, q1 - Controlled Quality (Quality Levels: 5 (highest), 3, 1 (lowest))

Table 3 Compression times (secs) corresponding to CR figures shown in Table 2. Note that all times are memory to memory on an 8 MHz IBM PC/AT.

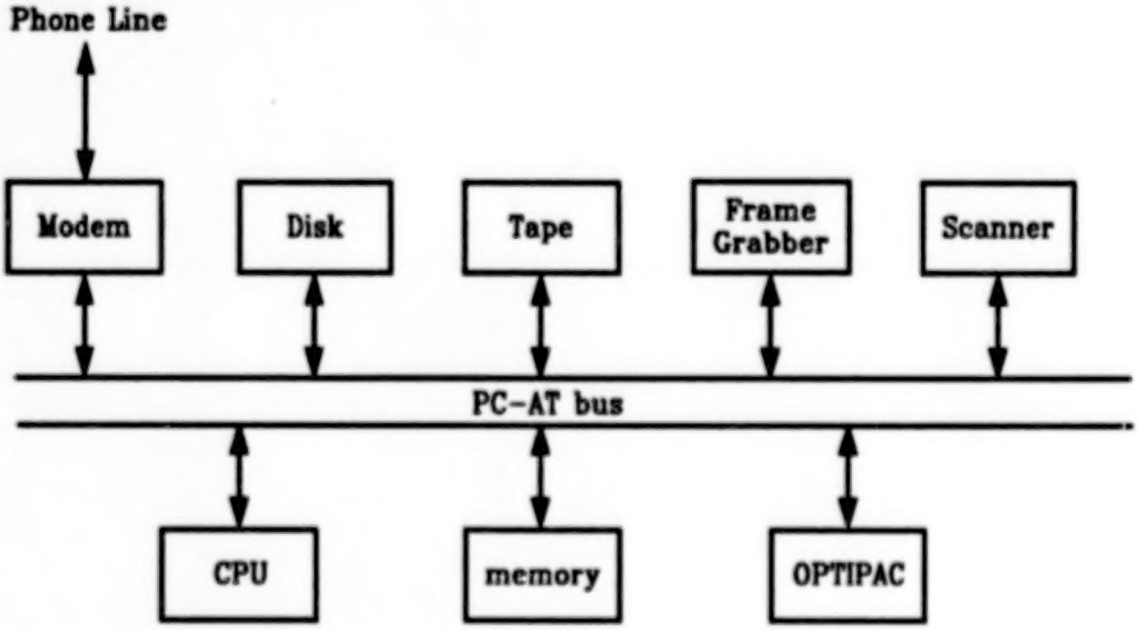


Figure 1 OPTIPAC™ in a PC environment. The input (or compressed) image from a disk, tape, image/document scanner, modem etc., is read into the main memory. It is then compressed (or expanded) by the OPTIPAC™ and routed to the output via main memory.

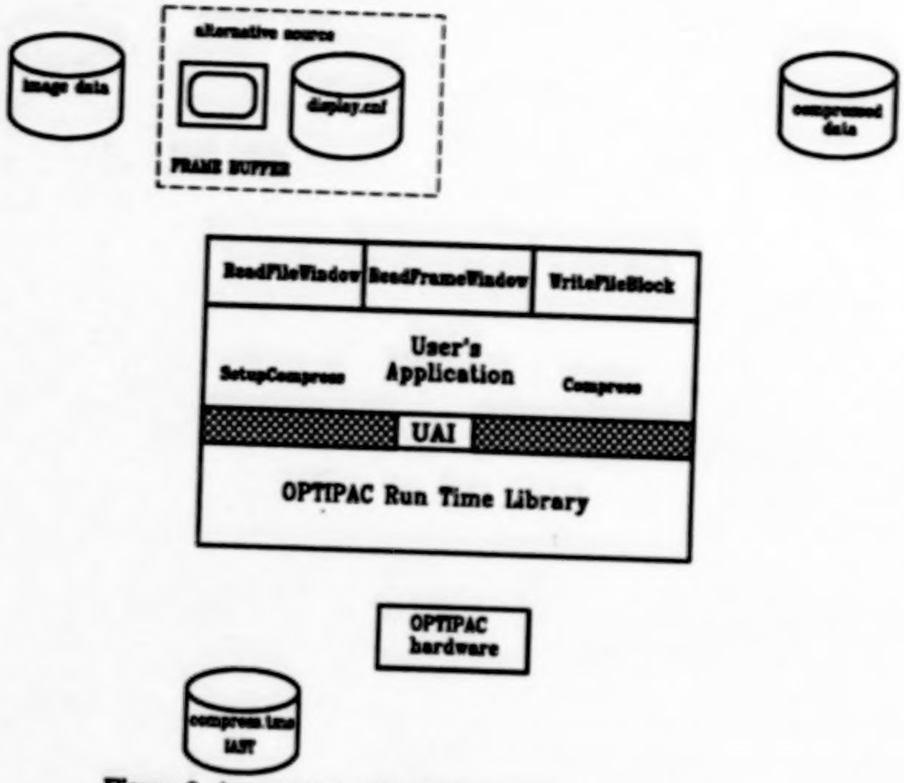

alternative source
image data
display.cal
FRAME BUFFER
compressed data
ReadFileWindow
ReadFrameWindow
WriteFileBlock
User's Application
SetupCompress
Compress
UAI
OPTIPAC Run Time Library
OPTIPAC hardware
compress.tms
IAST

Figure 2 An overview of the OPTIPAC™ compression system.

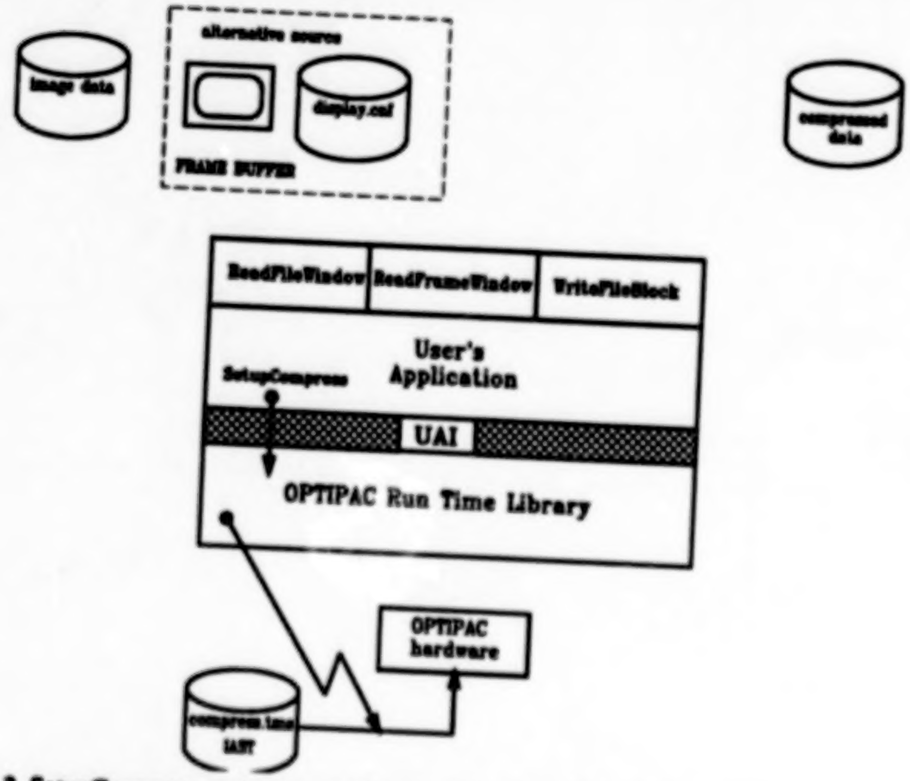

alternative source
image data
display.cal
FRAME BUFFER
compressed data
ReadFileWindow
ReadFrameWindow
WriteFileBlock
User's Application
SetupCompress
UAI
OPTIPAC Run Time Library
OPTIPAC hardware
compress.tms
IAST

Figure 3 SetupCompress - selecting and loading the appropriate compression algorithm from *compress.tms*, the DSP application library. Note that this data file also contains the IAST which is therefore independent of the application code.

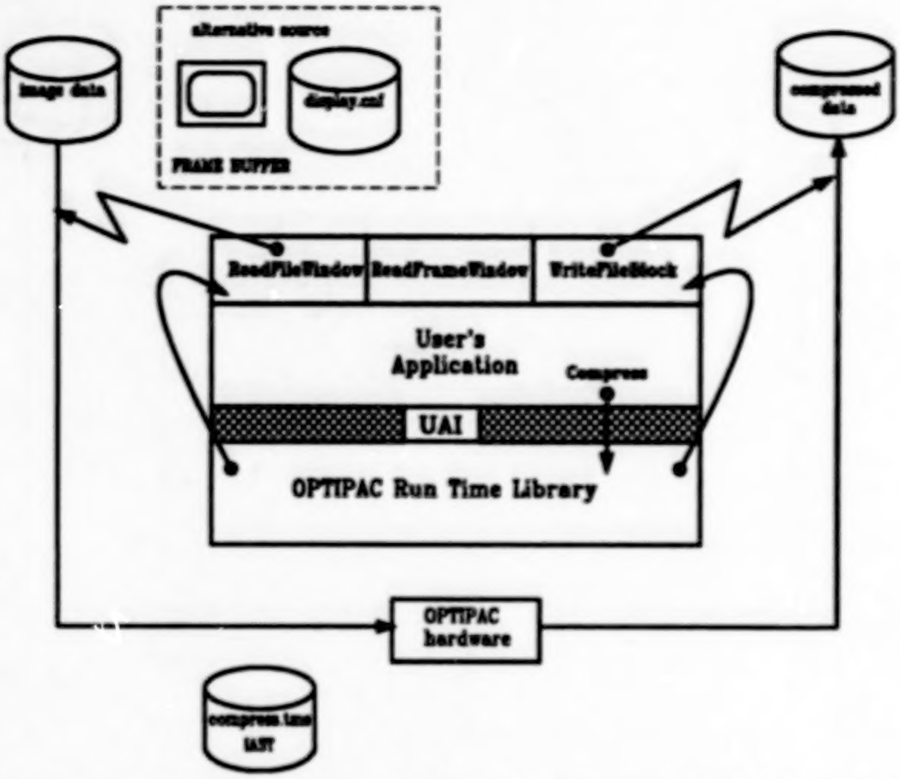

alternative source
image data
display.cnf
FRAME BUFFER
compressed data
ReadFileWindow
ReadFrameWindow
WriteFileBlock
User's Application
Compress
UAI
OPTIPAC Run Time Library
OPTIPAC hardware
compress.tme
IAST

Figure 4 Compress - compressing an image stored in a disk file to a second disk file. Note how the run time library uses I/O functions provided by the user to access image data. In this way any I/O device is readily accommodated.

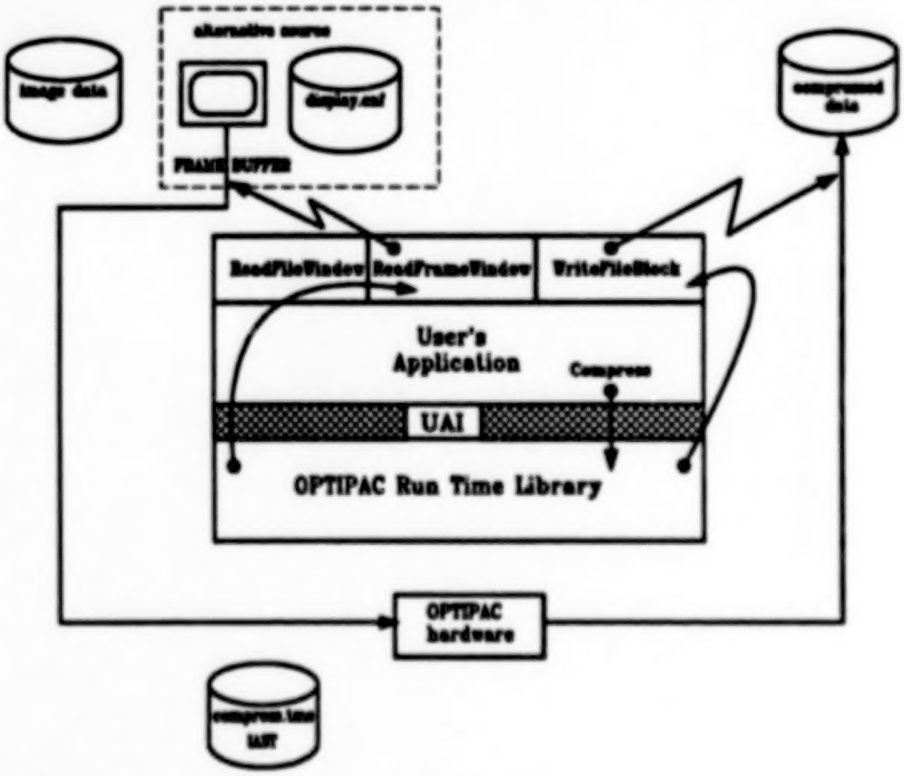

alternative source
image data
display.cnf
FRAME BUFFER
compressed data
ReadFileWindow
ReadFrameWindow
WriteFileBlock
User's Application
Compress
UAI
OPTIPAC Run Time Library
OPTIPAC hardware
compress.tme
IAST

Figure 5 Compress - compressing an image stored in a frame buffer to a disk file. Note the similarity with Fig. 4, the only difference is that *ReadFileWindow* has been replaced by *ReadFrameWindow*.

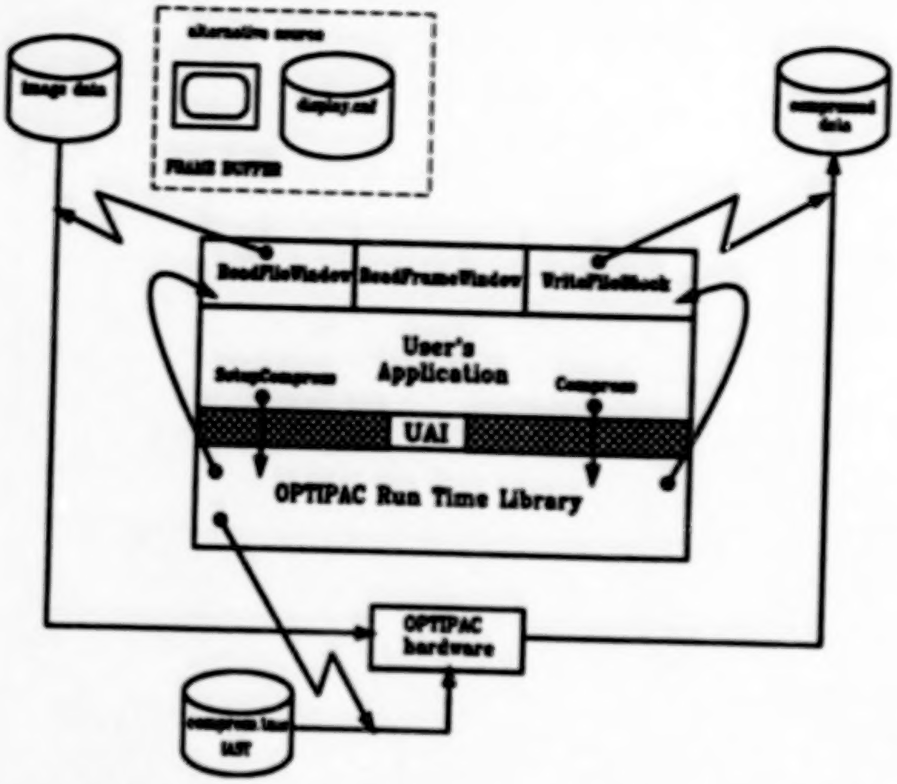


image data
alternative source
FRAME BUFFER
compressed data
ReadFileWindow
ReadFrameWindow
WriteFileBlock
User's Application
SetupCompress
Compress
UAI
OPTIPAC Run Time Library
OPTIPAC hardware

Figure 6 The complete OPTIPAC™ compression system showing the combined effects of the two stages: SetupCompress and Compress.

HYBRID LZW COMPRESSION

H. Garton Lewis Jr. and William B. Forsyth

Fairchild Weston Systems Inc.

Abstract:

The Science Data Management and Science Payload Operations subpanel reports from the NASA Conference on Scientific Data Compression (Snowbird, Utah in 1988) indicate the need for both lossless and lossy image data compression systems. The ranges developed by the subpanel suggest ratios of 2:1 to 4:1 for lossless coding and 2:1 to 6:1 for lossy predictive coding. For the NASA Freedom Science Video Processing Facility it would be highly desirable to implement one baseline compression system which would meet both of these criteria. This paper presents such a system utilizing an LZW hybrid coding scheme which is adaptable to either type of compression. Simulation results are presented with the hybrid LZW algorithm operating in each of its modes.

Introduction:

LZW lossless coding[1,2] is a completely reversible process; the encoding/decoding operations preserve all of the information contained in the input data sequence. This technique may be used on image data, computer files, or telemetered data. The hybrid system presented in this paper has the ability to compress image data in either a lossless or lossy mode, trading off data quality for data volume. It does this by means of an adaptive DPCM loop, which decorrelates the input image into symbols, before encoding with the LZW algorithm. The DPCM output symbols may be uniquely represented (lossless mode) or quantized (lossy mode.)

In addition, a mechanism is provided for the compression of non-video (telemetered) data. The mode of operation may be selected through the command and control system.

Decorrelating the input image (the decorrelation circuit):

Let S_i represent the value of the i^{th} element of an image vector **S**, being clocked out of a camera. The value of the previous pixel is thus S_{i-1}. The "error" vector (sometimes called the difference signal) **E,** is defined as $E_i=S_i-S_{i-1}$. A complete, alternate representation of **S** is the first pixel, $S_{0,}$ followed by the vector of error values, **E**. This representation contains enough information to reconstruct **S** (since $S_i = S_{i-1} + E_i$.) This idea forms the basis of predictive coding theory[3].

The compression ratio for lossless coding is bounded by the entropy of the input data. Since adjacent pixels in an image tend to be highly correlated, the resulting entropy for the vector **E** is much lower than that of the original vector **S**. **Fig. 1** shows a reconnaissance image (**S**) and its histogram, and **Fig. 2** shows the resulting decorrelated image (**E**) and its histogram. In **Fig. 2** all values of **E** have been made positive by adding an offset. Thus, E_i=0 is displayed at the middle of the graph.

Fig 1-1 (Recce. original)

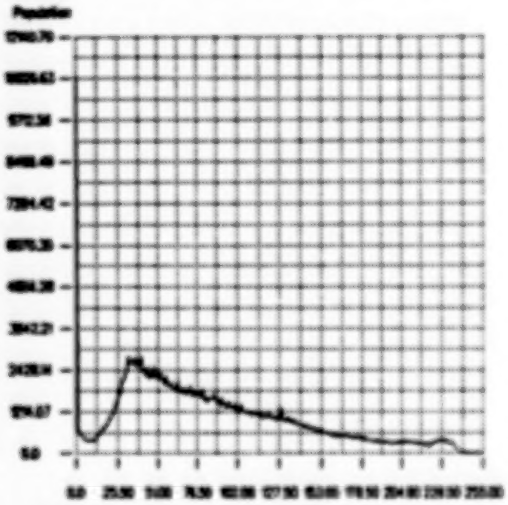


Fig 1-2 (Histogram of Recce. image)

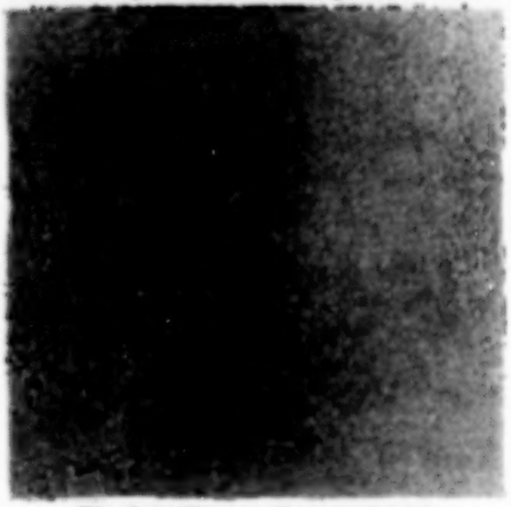


Fig 2-1 (Recce. Decorrelated)

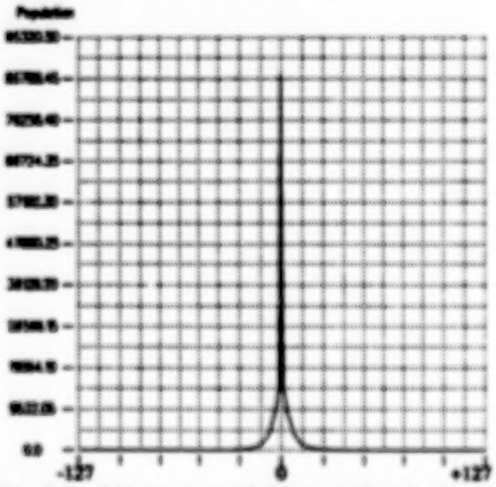


Fig 2-2 (Histogram of Decorrelated image)

Uniquely representing each error value, as in **Fig. 2**, requires 2^{n+1} bits, where n is the total number of bits per pixel in **S**. In this way, all information is preserved and **S** can be completely reconstructed; such a representation is used in the lossless mode of the hybrid LZW compressor.

Alternatively one can choose to quantize the error signal:

$$E_i' = \left\lfloor \frac{E_i + .5}{\Delta} \right\rfloor \Delta$$

where Δ = the current bin width of the quantizer

and then

$$S_i' = S_{i-1}' + E_i'$$

If $\Delta \geq 2$, then **E** can be represented with fewer bits, but quantization error is introduced into the image; this technique is used in the lossy mode of the compressor. The case Δ=1 is referred to as linear quantizing, and represents the lossless mode.

It is possible to switch between the lossy and lossless modes of operation by determining the representation to use to describe **E**: either uniquely representing or quantizing each E_i (see **Fig. 3**.) For the remainder of this paper, the error E_i', and error vector **E'**, will be used to represent output from the quantizer in **Fig. 3**.

Lossless - Linear Quantizer
Lossy - Non-Uniform Quantizer

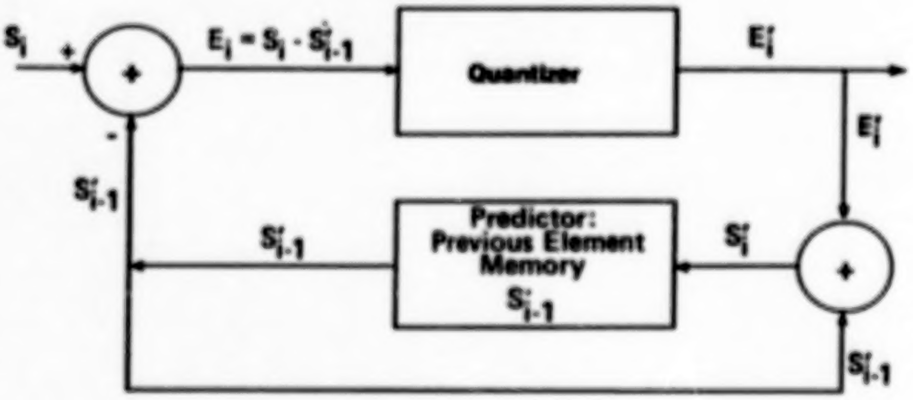


Fig 3. Decorrelation Circuit

Encoding the Error:

Once the quantized error vector **E'** is obtained, it is encoded with the LZW algorithm, thus preserving all information contained in **E'**. The LZW algorithm is a good choice since it approaches the lower bounds of compression ratios attainable by block-to-variable and variable-to-block codes designed to match specific source data. Since the LZW routine automatically adapts to changes in the source data no *a priori* information about **E'** is required.

System block diagram:

Fig. 4 shows the block diagram for the compression system. The image **S** (in **Fig. 4**, shown coming from an experiment occurring in the Space Station) passes through a subtractor which forms **E**. Then, depending on the mode selected, E_i' is formed by quantizing or singularly representing (linear quantizing) E_i.

The "predictor" block in **Fig. 4** is simply a function for improving the accuracy of the value E_i. Instead of representing E_i by S_i-S_{i-1}', some other equation may be used which takes into account the correlation between vertically adjacent pixels in the image. **S** is now represented as a two dimensional array $S_{i,j}$, where i and j have origin at 0 and are bounded by the maximum horizontal and vertical dimensions of the image. Determining good equations for E_i is the subject of many books and papers on predictive coding[4]. The equation used for the remainder of this paper is

$$E_i = S_i - P(i,j) \text{ ,}$$

where

$$P(i,j) = 0.75{*}S_{i-1,j}' - 0.5{*}S_{i-1,j-1}' + 0.75{*}S_{i,j-1}'$$

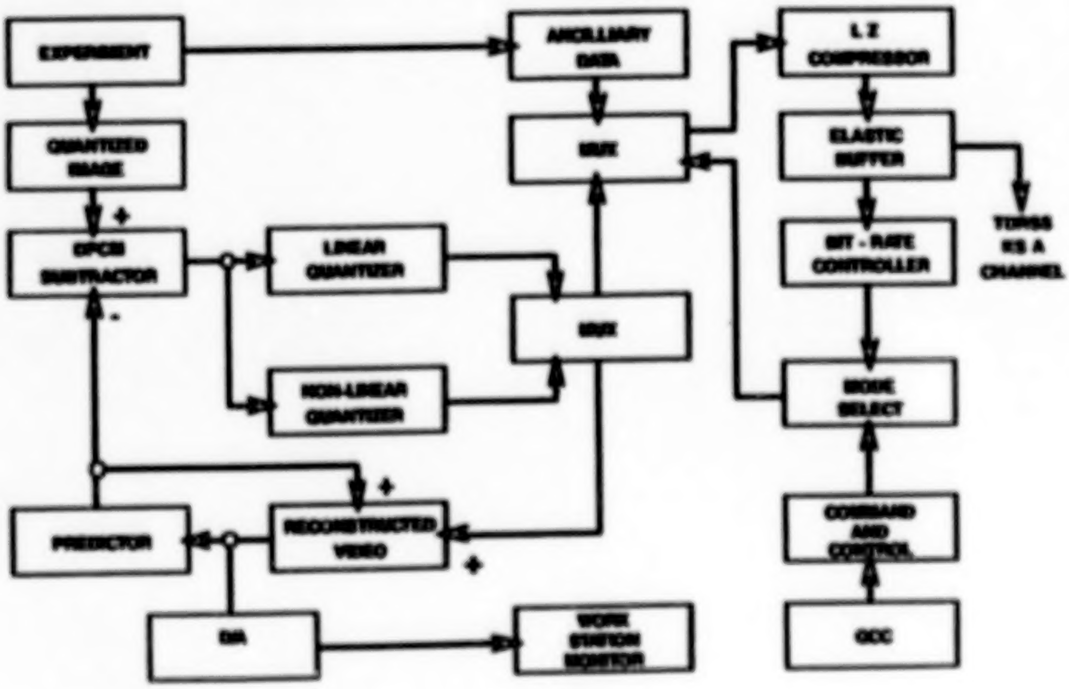

EXPERIMENT
ANCILLARY DATA
L Z COMPRESSOR
QUANTIZED IMAGE
MUX
ELASTIC BUFFER
+
DPCM SUBTRACTOR
LINEAR QUANTIZER
BIT - RATE CONTROLLER
TDRSS KS A CHANNEL
-
MUX
NON-LINEAR QUANTIZER
MODE SELECT
+
PREDICTOR
RECONSTRUCTED VIDEO
+
COMMAND AND CONTROL
D/A
WORK STATION MONITOR
GCC

Fig 4. Hybrid LZW system block diagram

After quantizing, the signal **E′** is fed into the LZW compressor which codes it without loss. The resulting symbols are placed into an elastic buffer (described later), and are finally output into a fixed channel (in **Fig. 4**, the TDRSS KSA channel.)

The elastic buffer and bit-rate controller are shown as two separate blocks (in reality they are often merged.) The purpose of the bit-rate controller is to switch between lossless and lossy modes so that the output symbols from the LZW compressor are of constant rate (in bits/pixel) on average. The bit-rate controller does this by examining the current rate of compression and changing the current mode of operation (lossless or lossy), when necessary, to bring the rate into the desired range.

Additional control is provided by resetting the LZW symbol table when compression falls below the channel output rate. In such a case, symbols are being generated too efficiently. By resetting the LZW table, we force the algorithm to begin learning the scene statistics anew. This results in a lower compression rate until the table fills up.

Unfortunately, the rate out of the LZW encoder cannot be controlled at every instant. Thus, the elastic buffer is needed to hold several lines of image information in case of drastic changes in the scene statistics (which cause a large change in the instantaneous bit-rate.) In this way, constant output rate may be assured over the entire image, if not on a line by line basis.

In addition to compressing image data, the system provides a path for non-image data. This ancillary information (such as data from experiments) is fed directly into the LZW compressor.

Results:

A variety of images were selected to examine the performance of the hybrid LZW compression system in each of its modes. Some results for the lossy mode ($\Delta=8$) are shown in **Fig. 5**. Two of the images, (Docking Target and Satellite) contain significant amounts of tape noise. Lossless results are not shown since the original image is always equal to the reconstructed image. **Tables 1** and **2** show compression of between 3.4 and 4.3 bits/pixel for the lossless mode, and 1.24 to 1.59 bits/pixel in the lossy mode with $\Delta=8$. The quantizer used for the lossy case is a non-uniform quantizer (**Fig. 6**.) For such a quantizer, the width of each bin increases as E_i' increases, and the output of each bin is not its center, but rather its centroid.

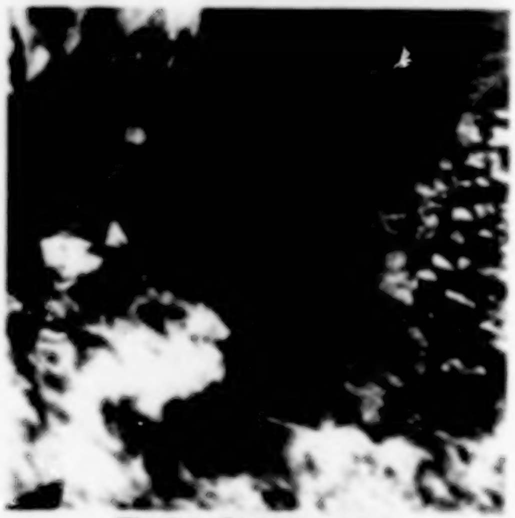


Fig 5-1. Satellite Image

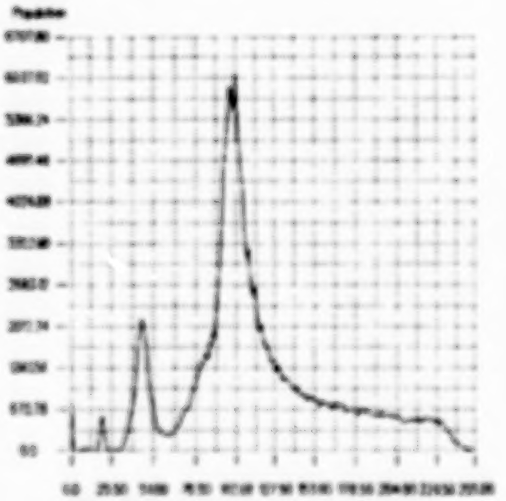


Fig 5-2. Histogram of Satellite

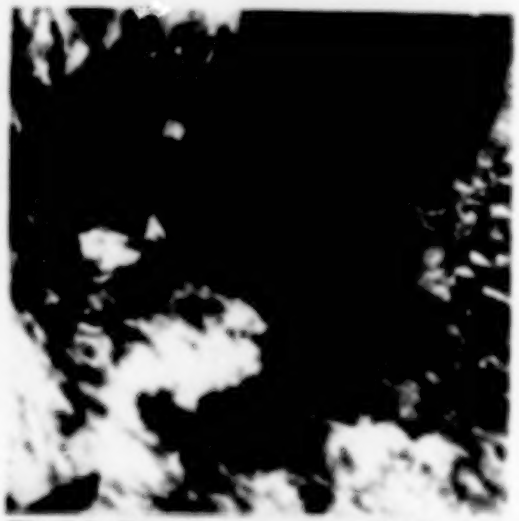


Fig 5-3. Satellite reconstructed with $\Delta=8$

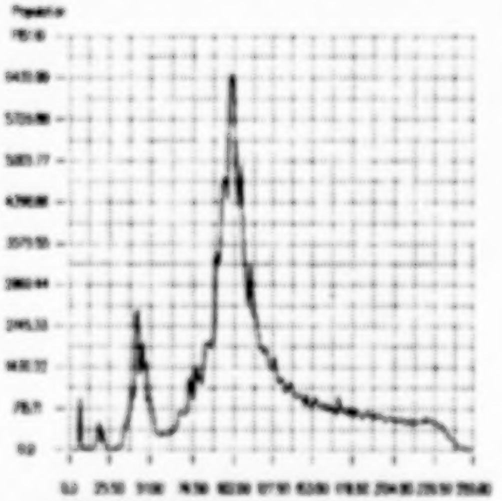


Fig 5-4. Histogram of Fig 5-3.

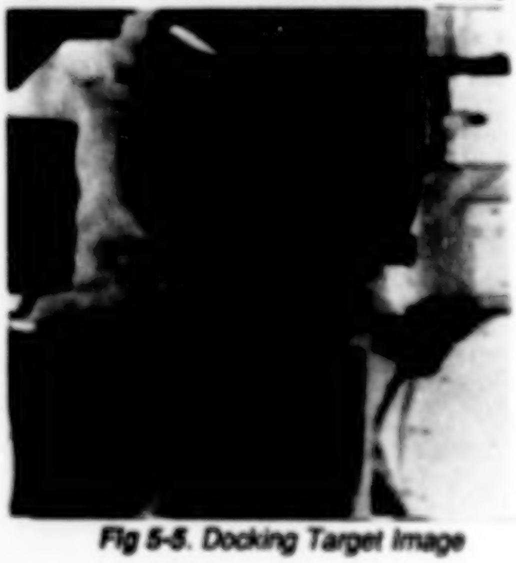


Fig 5-5. Docking Target Image

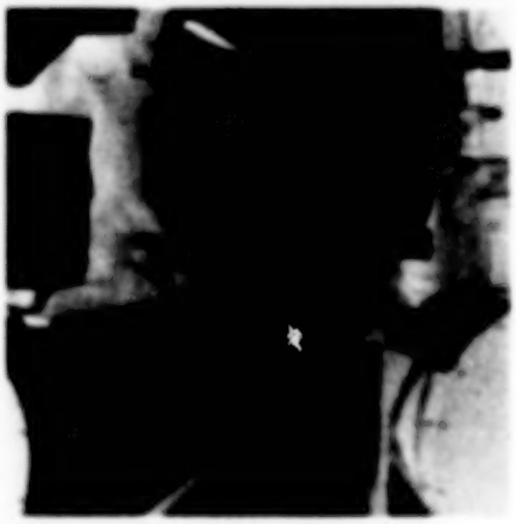


Fig 5-6. Dock. Targ. reconstructed with $\Delta=8$

Lossless L Z Compression

IMAGE	ENTROPY ORGIONAL IMAGE BITS/PIXEL	LINEAR DECORRELATED ENTROPY BITS/PIXEL	L Z COMPRESSION BITS/PIXEL	L Z COMPRESSION RATIO
DOCKING TARGET	7.31	3.61	3.40	2.05
SATELLITE	7.23	4.05	4.3	1.86
CLOUDS	6.38	3.41	3.74	2.13

Table 1. Lossless Compression Results

Non - Uniform Quantizer - Followed By L Z Encoder

IMAGE	ENTROPY ORGIONAL IMAGE BITS/PIXEL	NON-LINEAR DECORRELATED ENTROPY BITS/PIXEL	L Z COMPRESSION BITS/PIXEL	L Z COMPRESSION RATIO
DOCKING TARGET	7.31	1.18	1.24	6.45
SATELLITE	7.23	1.54	1.59	5.03
CLOUDS	6.38	1.24	1.30	6.15

Table 2. Lossy Compression Results

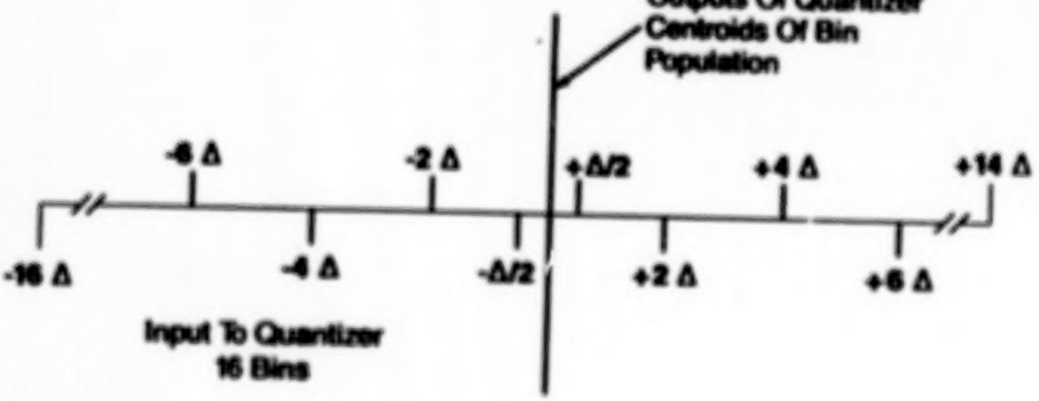

Outputs Of Quantizer
Centroids Of Bin
Population
-6 Δ
-2 Δ
+Δ/2
+4 Δ
+14 Δ
-16 Δ
-4 Δ
-Δ/2
+2 Δ
+6 Δ
Input To Quantizer
16 Bins

Fig 6. Non-uniform quantizer

Future Plans:

Current results from the hybrid LZW routine indicate that it is possible to satisfy both lossless and lossy requirements in a single system. In order to gain even greater compression ratios, several possible enhancements are currently being investigated.

Improved control of the bit-rate out of the quantizer is being developed for the lossy case. Control is provided by modification of Δ on a line by line basis, that is, the current bit-rate is examined at the end of every line, and Δ is adjusted (if necessary) to raise or lower the output symbols to the desired rate.

Further optimization of the non-uniform quantizer is possible. A constrained loss quantizer should be implemented. This quantizer uses a $\Delta=1$ (linear quantizer) for low values of E_i, and uses larger values of Δ as the values of E_i increase.

Finally, use of the Discrete Cosine Transform to decorrelate **S** instead of a DPCM predictor is being considered. LZW compression ratios of 16:1 have been realized for DCT coefficients.

Summary:

A hybrid LZW compression system has been presented which satisfies NASA requirements for both lossy and lossless compression. The system has been simulated on a computer, and has produced good quality output images at rates as low as 1.24 bits/pixel. The system appears to have applicability for spacecraft and reconnaissance imagery.

References:

[1] A. Lempel and J. Ziv. "A universal algorithm for sequential data compression." *IEEE Trans. Inform. Theory*, IT-23(3):337-343, May 1977.

[2] Terry A. Welch. "A Technique for High-Performance Data Compression." *Computer*, pp. 8-19, June 1984.

[3] Arun N. Netravali and Barry G. Haskell. *Digital Pictures Representation and Compression*. Plenum Press, New York, 1982.

[4] Azriel Rosenfeld and Avinash C. Kak. *Digital Picture Processing*. Academic Press, New York, 1988.

CONNECTIONIST MODEL-BASED STEREO VISION FOR TELEROBOTICS

William Hoff and Donald Mathis
Martin Marietta Astronautics Group

SUMMARY

Autonomous stereo vision for range measurement could greatly enhance the performance of telerobotic systems. Stereo vision could be a key component for autonomous object recognition and localization, thus enabling the system to perform low-level tasks, and allowing a human operator to perform a supervisory role. The central difficulty in stereo vision is the ambiguity in matching corresponding points in the left and right images. However, if one has *a priori* knowledge of the characteristics of the objects in the scene, as is often the case in telerobotics, a model-based approach can be taken.

In this paper, we describe how matching ambiguities can be resolved by ensuring that the resulting three-dimensional points are consistent with surface models of the expected objects. A four-layer neural network hierarchy is used in which surface models of increasing complexity are represented in successive layers. These models are represented using a connectionist scheme called *parameter networks*, in which a parametrized object (for example, a planar patch $p=f(h,m_x,m_y)$) is represented by a collection of processing units, each of which corresponds to a distinct combination of parameter values. The activity level of each unit in a parameter network can be thought of as representing the confidence with which the hypothesis represented by that unit is believed. Weights in the network are set so as to implement gradient descent in an energy function.

1. INTRODUCTION

The goal of autonomous stereo vision is to determine the three-dimensional distance, or depth, of points in a scene from a stereo pair of images. This information is very useful for higher level perception capabilities such as autonomous object recognition and localization. Such capabilities could greatly enhance the performance of telerobotic systems, by enabling the system to perform low-level tasks and allowing the human operator to perform a more supervisory role. This paper describes a new approach to stereo vision that we have implemented and with which we have obtained preliminary results. Section 2 provides an overview of stereo vision and our general approach. Section 3 describes our implementation of the approach with a connectionist, or neural network model. Sections 4 and 5 give additional details of the implementation. Section 6 describes results, and Section 7 gives conclusions.

2. STEREO VISION

The usual approach to stereo vision includes the following steps [1]: (1) Features are extracted from each image independently, (2) features from one image are matched with the corresponding features from the other image, thus deriving their depths, and (3) a surface is interpolated between the possibly sparse depth points. The central difficulty in autonomous stereo vision is the second step; *i.e.*, matching corresponding points in the left and right images [2]. The reason is that matching is highly ambiguous, since there is little information to characterize low-level features (*e.g.*, edge points) uniquely. Although less ambiguous higher level features such as long line segments [3] can be used, these approaches are less general and do not explain the ability of the human visual system to fuse stereograms consisting solely of random dots [4].

A different approach to solving the correspondence problem, which we have taken here, is to use a model-based approach, which requires some *a priori* knowledge of the characteristics of the objects in the scene. Specifically, parametrized surface models are defined, such as planar and quadratic patches, and only those matches are allowed that are consistent with those models. The surfaces of the objects in the scene are assumed to be composed of a set of these patches. Low-level features (*i.e.*, edge points) are used, which makes the approach general in the sense that no specific markings or texture patterns are required. However, the specific surface models restrict the applicability of the approach to situations where the objects in the scene are indeed describable by these surface models. This is a reasonable approximation in most cases, especially for the man-made objects which would be encountered in many situations in telerobotics.

To use this surface-based approach, the processes of matching and surface interpolation must be integrated. This is because matching provides surface depth values at the locations of the matched features, which constrain the interpolated surface. However, the correctness of the choice of matches is judged by the type of surface produced. Therefore, the interpolation process should be integrated with matching so that acceptable matching decisions can be made.

A stereo vision algorithm that integrated matching and surface interpolation was recently developed by one of the authors (Hoff) [5]. Similar approaches have been taken by other researchers [6-8]. The work described in this paper primarily differs from these other approaches in that it uses a connectionist implementation. This approach tightly integrates matching and surface interpolation and naturally combines top-down and bottom-up processing.

3. CONNECTIONIST IMPLEMENTATION OF STEREO CORRESPONDENCE

In connectionist or neural network architectures, a large number of simple processing units are connected by weighted connections. In our stereo implementation, each unit represents a hypothesis about a parametrized model; *i.e.*, a feature correspondence or a surface patch. Each unit has an activation value, which represents the confidence of the hypothesis. Ballard calls this class of models parameter networks [9]. Positive and negative connections between pairs of units represent consistency constraints between the hypotheses they represent. The units then incrementally update their activation values in parallel, based on the activation values of their neighbors and the connections to them. It is this parallel computation which can be used to tightly integrate matching and surface interpolation.

4. DETAILED DESCRIPTION OF THE MODEL: STRUCTURE

4.1 Hierarchical Structure of the Model

Our model consists of a four-layer neural network hierarchy, in which features of increasing complexity are represented in successive layers of the hierarchy. The model takes two stereo images as input at the lowest layer, and produces surface depth and orientation estimates in the highest layer (fig. 1).

Level 1.

The first level of the hierarchy consists of two stereo images that have been preprocessed to detect edge point features. It is assumed that the images are of a densely textured surface, so that there is an abundance of edge features in the images. As far as our model is concerned, the details of the edge-detector are not important. (Note: Unlike other approaches [10], the performance of this algorithm is not tied directly to the spatial scale of the edge detector.)

Level 2.

The second layer consists of representations of correspondences between edge point features in the left and right images. Each correspondence has an associated stereo disparity value, from which the actual 3D position of the point can be calculated. Since we are using local representations, one unit is allocated for each possible depth (or disparity) at each pixel location. (Note: the 3D-points and surface patches are all represented in the coordinate system of the left image.) The range of allowable disparities is a parameter of the model, and is the same at all pixel locations[1].

Level 3.

---

[1] This is a deficiency of the present model compared to those that allow different disparity estimates at different locations. In the present model, to allow for a wide range of disparities across an image, one must allocate units to account for all possible disparities at every pixel location, regardless of the particular disparity estimate at that location.

The third layer consists of representations of overlapping surface patches of constant depth. Each patch covers a region in 3D space of a fixed diameter (in pixels, not in physical space), and at a constant depth. The representation of the entire surface at this level of the hierarchy would consist of a surface patch at each (x,y) location, each patch being a little plane of zero slope and some known depth. In this level the spatial resolution is reduced so that not every (x,y) pixel location contains an estimate. The diameter of the surface patches is a parameter of the model, and is the same for all patches and at all pixel locations. The range of allowable depths is the same as that of level 2.

Level 4.

The fourth layer consists of representations of surface patches of known depth and constant (but possibly non-zero) slope. These patches cover a larger spatial region than the constant-depth patches, and they also overlap spatially. The representation of the entire surface at this level of the hierarchy would consist of a surface patch at each (x,y) location, each patch being a little plane of some known slope and some known depth. The difference between level 3 and level 4 is that the level 4 patches can have non-zero slope. Again, the spatial resolution is reduced. Three parameters are used to represent a surface patch of non-zero slope: ∂, m_x and m_y - the depth, and the x and y components of the slope. One unit is used to represent each (∂, m_x, m_y) combination.

Additional Levels.

Additional levels could be defined, such as higher order surface patches (*e.g.*, quadratic surfaces) or volumetric primitives. Higher levels would represent more specialized descriptions and would implement more powerful matching constraints.

4.2 Connections in the Model

Since we have designed the units in the model to represent specific things, we can also design the connections between them. There are two types of connections: (1) Inhibitory, to implement the competition between mutually inconsistent units, and (2) excitatory, to implement the support that mutually compatible units can give to each other. All connections in the model are bi-directional, so the model contains feedback of activation from higher layers to lower layers. Thus, the model can be seen as an instance of an interactive activation and competition model [11].

Inhibitory Connections.

Since separate units are used to represent each possible surface hypothesis at each pixel location, they represent *conflicting estimates* or *inconsistent hypotheses* of the surface at that point. We would therefore like these units to compete against each other, and so at each level, the units located at the same (x,y) position are fully connected with inhibitory connections. In level 4, this means that each of these 3D grids of units forms a giant competing pool of units - there can be only one winning (∂, m_x, m_y) combination at a given patch location.

Excitatory Connections.

To set excitatory connections, units are connected to other units that support each other. Each level 2 unit represents a 3D-point with a particular disparity (∂) at a particular (x,y) location. As shown in figure 2, it receives input from level 1 from the two locations that it accounts for — one at (x,y) in the left image, and one at $(x+\partial,y)$ in the right image.

Each level 3 unit represents a surface patch at a particular constant depth ∂ and location (x,y), and receives input from all 3D-point units (in level 2) in a local region that lie sufficiently close to the (x,y) location and (∂) depth represented by the level-3 unit (see fig. 3). Each level 4 unit represents a surface patch at a particular depth ∂, location (x,y), and slope (m_x,m_y) and receives input from all level-3 units that lie sufficiently close to the (x,y) location, (∂) depth, and (m_x,m_y) slope represented by the level-4 unit (see fig. 4).

The size of the support regions imply a characteristic scale assumption in the model. The support widths are parameters of the model. For example, a *support-region width* of *w* pixels from level 2 units to level 3 units embodies the assumption that the surface is smooth at the scale of *w* pixels. The disparity range and surface slope range indicate assumptions about the allowable ranges for distance and orientation of the visible surfaces.

5. DETAILED DESCRIPTION OF THE MODEL: WEIGHTS

Now that we have the structure of the model — *i.e.*, the layers, units, and connections are defined — we must assign weights to the connections. This is not a trivial problem, since it is difficult to specify in advance the relative importance of the various components, yet the performance of the model depends critically on setting the weights properly. One possibility is to have the network learn the weights on its own. This introduces additional problems, however, although we would like to explore this approach eventually. For example, one must create a training set of data that covers the important examples, and in the appropriate proportions. Also, training a network with feedback is computationally expensive [12].

Instead, we have chosen to set the weights so as to implement the minimization of an energy (or "cost") function. Specifically, the weights implement gradient descent in this energy function [13]. This increases the likelihood of the system settling into a stable state, and a state that satisfies the requirements we wish to impose on the system (e.g., winner-take-all behavior, etc). The use of an energy function also greatly increases our ability to analyze the model and make meaningful adjustments to its parameters.

The basic idea is that an energy (or "cost") function is defined as a function of the activations of all the units in the network, and this energy function is designed to be minimized when the pattern of activity in the network corresponds to a good solution to the problem the network is trying to solve. Each unit then locally computes the gradient of the energy function with respect to its own activation, and adjusts its activation in the direction that reduces the energy. This process is repeated until the network settles on a stable pattern of activity.

There are many variations of gradient descent, including the IAC model, the BSB model [14], and Grossberg's model [15]. We used a very simple activation function:

$$a_j(t+1) = a_j(t) + \Delta a_i(t), \qquad (1)$$

where $\Delta a_i(t) = -\varepsilon * grad_i \bullet (1 - a_i(t)) \quad \text{if } grad_i \leq 0$

or $\Delta a_i(t) = -\varepsilon * grad_i \bullet a_i(t) \quad \text{if } grad_i > 0$

Here, *grad*$_i$ is the rate of change of energy with respect to a_i, or $\partial E/\partial a_i$. This is based on gradient descent, but is not exactly the same as gradient descent since the activations (a_i) are bounded by 0 and 1.

5.1 The Energy Function

There are many different energy functions that can be used to implement our model. Some energy functions contain more terms than others, and some seem to be easier to understand than others. We made the decision to try to minimize complexity, while still using a function that would provide good results. The energy function that was used is a sum of six terms. Each term represents a constraint on good solutions to the surface estimation problem:

$$E(\mathbf{a}) = \qquad (2)$$

$$- \sum_{i \in L2} e_i a_i \qquad \text{(Term 1: Image Evidence)}$$

$$+ k1 \sum_{c \in L2} \; \sum_{i \in c} \; \sum_{j \in c \; j \neq i} a_i a_j \qquad \text{(Term 2: WTA - Level 2)}$$

$$- k2 \sum_{i \in L2} \sum_{j \in S_i^{L3}} a_i a_j f_{23}(i,j) \qquad \text{(Term 3: L2} \Leftrightarrow \text{L3 support)}$$

$$+ k3 \sum_{c \in L3} \sum_{i \in c} \sum_{j \in c\; j \neq i} a_i a_j \qquad \text{(Term 4: WTA - Level 3)}$$

$$+ k4 \sum_{c \in L4} \sum_{i \in c} \sum_{j \in c\; j \neq i} a_i a_j \qquad \text{(Term 5: WTA - Level 4)}$$

$$- k5 \sum_{i \in L4} \sum_{j \in S_i^{L3}} a_i a_j f_{34}(i,j) \qquad \text{(Term 6: L3} \Leftrightarrow \text{L4 support)}$$

where the function *f*(i,j) is a "closeness" function used to weight the support of a point *i* for a plane *j* within a threshold distance "d" by an amount inversely proportional to the point's distance from the plane:

$$f(i,j) = 0 \qquad \text{if distance(point(i), plane(j))} > d \qquad (3)$$

$$\text{or} \qquad f(i,j) = \frac{d - \text{distance(point(i), plane(j))}}{d} \qquad \text{otherwise.}$$

The energy function (Eq. 2) has six terms, but there are actually only three different types of terms used. The first type of term (term 1) is minimized when the level 2 units maximally "agree" with the image evidence. The second type of term is minimized when winner-take-all situations exist wherever they are desired. There are three terms of this type (terms 2,4,5), one for each layer in which WTA situations are desired (the "c" parameter ranges over all competitive pools in the layer). The third type of term is minimized when hypotheses in one layer maximally "agree" with related hypotheses in adjacent layers (terms 3,6). The "S" variables are the units' "support regions" (*e.g.*, S_i^{L3} is the set of units in Level 3 that support unit *i*..

The constants k_i are used to represent the relative importance of the various terms within the overall energy function. By inspection, it can be seen that the WTA terms are minimized when no more than one unit has a nonzero activation. The "support" terms are all functions of a single unit in one layer and a group of units in its "support region" in an adjacent layer. These terms are minimized when the unit with the most support within a competitive pool is the winner in that pool. The "Image Evidence" term (term 1) is minimized when the 3D point unit with the most support within a competitive pool is the winner in that pool.

5.2 Connection Weights

The connection weights in the network are derived from the energy function by taking derivatives with respect to the unit activations. In order for a unit to be able to calculate the derivative locally, it usually needs to know the activations of some other set of units. This communication is implemented with weighted connections between the units. For units in level 2 of the present model, this derivative takes the form:

$$\text{grad}_i = \frac{\partial E}{\partial a_i} = k1 \sum_{j \in c_i} a_j - (k2 + k3) \sum_{k \in S_i^{L3}} a_k f_{23}(i,j) - e_i . \qquad \text{(with } j \neq i\text{)} \qquad (4)$$

where c_i is a "competitive pool" of units of which unit *i* is a part, and S_i^{L3} is the "support region" in Level 3 for unit *i* .

The unit *i* can compute this derivative if we create connections between it and all the other units that appear in the above equation. The unit can then calculate the gradient using a net input function such as:

$$net_i = \sum_j w_{ij} a_j - e_i \qquad (5)$$

where the subscript *j* ranges over all of the units to which unit *i* is connected. The term e_i is the "external input" to the unit, which in this case is the input from the edge images. To make this technique work, however, the weights (the w_{ij}'s) must be set to the following values:

$$w_{ij} = k1, \qquad \text{for all units } j \in c_i, \text{ with } j \neq i. \qquad (6)$$

$$w_{ik} = -k2 f_{23}(i,j), \qquad \text{for all units } k \in S_i^{L3}.$$

To put it another way, the connectivity and the connection weights themselves are derived from the energy function by first deriving the equation for $\partial E/\partial a_i$ from the energy function, and then choosing a set of connections and a net input function that *implement* the $\partial E/\partial a_i$ equation. This technique was used to determine the connectivity and weights throughout the network.

Note that the energy function still contains 5 unknown parameters (the k_i.), which must be selected by trial and error. However, we can derive some useful relationships between these parameters, by strictly enforcing the winner-take-all requirement. For example, suppose that *C* is a set of units that forms a competing pool (*i.e.*, a "winner-take-all" pool). If the current pattern of activations of the units in *C* is a winner-take-all pattern (i.e., there is one unit with a "1" activation, and all of the others have a "0" activation), then this activation pattern will *remain* stable if:

$$grad_w \leq 0 \qquad \text{for the "winning" unit } (w \in C), \text{ and} \qquad (7)$$

$$grad_l \geq 0 \qquad \text{for all the "losing" units } (l \in C).$$

This is true because in gradient *descent*, the activation values are changed in a direction opposite in sign from that of the gradient, so if the above conditions hold, the winning unit will try to *increase* its activation (but will be limited at 1), and the losing units will try to *decrease* their activations (but will be limited at 0). Therefore the winner-take-all pattern is stable[1].

When these conditions are applied to the competing pools in our model, we get the following results:

$$k1 \geq (k2)W_{2,3} + e_i \qquad (8a)$$

$$k3 \geq (k2)W_{2,3} + (k5)W_{3,4} \qquad (8b)$$

$$k4 \geq (k5)W_{3,4} \qquad (8c)$$

where $W_{a,b}$ is the width of the support region between units in Level a and units in Level b. So given values for k2 and k5, we can set the other k values according to these equations and be sure that all winner-take-all patterns will be stable.

6. RESULTS

The system was run primarily on 1-dimensional random-dot stereograms. (A 1-d stereogram image is just a single horizontal row of binary pixels). The system was run many times with different parameter settings to study the effects of enforcing WTA behavior to varying degrees. It was found that if the *k* parameters were set to values that would strictly enforce WTA, the network settled on *very poor* solutions to the surface estimation problem. Better results were achieved when the parameters governing the lower levels were set much lower than that required to enforce WTA (*i.e.*, less than half the value), and the higher-level parameters were set just slightly lower than that required for WTA. The best results were achieved when, instead of setting the parameters to fixed values, they were all set initially to values far below the WTA values, and then slowly increased to the WTA values. This appeared to work better because it allowed partial activation of large numbers of "almost right" hypotheses initially, and then as the parameters were raised, the best hypotheses were "selected" (more or less) from the set of partially active hypotheses.

---

[1] Note that this does not ensure that the network will *settle* on a winner-take-all pattern, but only that *if* a winner-take-all pattern exists, it will remain stable.

How to interpret the graphics:

Each diagram consists of three large rectangles filled with small squares. The highest rectangle represents the 3d-point layer of the network, the next lower rectangle represents the constant-depth-patch layer, and the lowest rectangle represents the constant-slope layer. Each rectangle is a graphic representation of the activations of the units in that layer of the network.

The 3d-point layer of the net is a 2D parameter space, (*x* vs. *disparity*). The horizontal dimension is *x*, (it ranges from x=0 to x=50), and the vertical dimension is *disparity* (-5 to +5). Each point in the space contains a unit, and the size of the black square at that point represents the activity of the unit. The constant-depth layer also has these dimensions. The constant-slope layer of the network has an extra dimension: *slope*. This extra dimension is graphically displayed as another level of rectangles-within-rectangles. The large rectangle contains of a horizontal row of "tall" rectangles. There is one of these for every other *x* value (the spatial resolution was reduced by a factor of 2). Each of these tall rectangles is filled with 77 tiny rectangles, each of which represents a distinct combination of (*x*, *disparity*, and *slope*). These tiny rectangles are arranged in a 2D grid, the vertical dimension of which is disparity (just as it is in the lower layers), and the horizontal dimension is *slope*. The slope ranges from -0.6 to +0.6, in increments of .2.

Figures 5a-5c depict the history of one run of the network after 1, 5 and 9 iterations. The input to the network is a stereo pair of images of a surface with slope -0.2 (this would appear as a diagonal line running from the top-left to the bottom-right of the large rectangles. Within the large "tall" rectangles, a slope of -0.2 would appear as a filled-in tiny rectangle three from the right (slope increases right to left; zero slope is in the center). By the end of the series, you can see that the disparity has been estimated fairly well. The slopes are roughly correct throughout the image (± .2), and they are exactly correct at 15 of the 25 pixels, or in 60% of the image).

7. CONCLUSIONS

The system performed very well on the 1D stereogram data. Addition of the second spatial dimension will require an extension of the support regions into the *y* dimension, but will not require a change to the competitive pools, since they extend only in the x direction. The energy function itself will not change (except for the fact that the S_i's will change). The changes will be primarily *quantitative*, not *qualitative*. An improvement in performance is expected, however, since the extension of the "cooperative" constraints (smoothness, planarity) into the y dimension will allow 3d-point hypotheses to take into account the smoothness (or lack thereof) of the surface in the *y* dimension. There will be an impact on the derivations of the *k*-values needed to preserve WTA, since there will be more competitive pools in the support region of a given hypothesis.

It is still not known exactly how the values of the *k* coefficients should be changed over time to maximize performance. But since the network is trying to find an energy minimum (ideally, the *global* minimum), simulated annealing with a fixed set of *k*'s would probably produce equal or (more likely) *better* results that the "pseudo-annealing" that was used here.

Despite the attempts to derive the values of as many of the model's parameters as possible, some amount of "rapid prototyping" was still ultimately necessary. For example, there was little guidance in picking the relative values of the "support weight" parameters (k2 and k5). For our purposes, we made them equal.

REFERENCES

[1] Barnard, S. and M. Fischler, "Computational Stereo," *ACM Computing Surveys*, Vol. 14, No. 4, Dec 1982, pp. 195-210.

[2] Marr, D., *Vision*, Freeman, San Francisco, 1982.

[3] Medioni, G. and R. Nevatia, "Segment-based Stereo Matching," *Computer Vision, Graphics, Image Processing*, Vol. 31, July 1985, pp. 2-18.

[4] Julesz, B., *Foundations of Cyclopean Perception*, University of Chicago Press, Chicago, 1971.

[5] Hoff, W. and N. Ahuja, "Surfaces from Stereo: Integrating Feature Matching, Disparity Estimation, and Contour Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 11, No. 2, February 1989, pp. 121-136.

[6] Olsen, S., "Concurrent Solution of the Stereo Correspondence Problem and the Surface Reconstruction Problem," *Proc. Eighth Int. Conf. Pattern Recognition*, 1986, pp. 1038-1041.

[7] Eastman, R. and A. Waxman, "Using Disparity Functionals for Stereo Correspondence and Surface Reconstruction," *Computer Vision, Graphics, Image Processing*, Vol. 39, 1987, pp. 73-101.

[8] Boult, T. and L. Chen, "Synergistic Smooth Surface Stereo," *Proc. IEEE Second Int. Conf. Computer Vision*, Dec 1988, pp. 118-122.

[9] Ballard, D., "Parameter Nets," *Artificial Intelligence 22*, 1984, pp. 235-267.

[10] Marr, D. and T. Poggio, "A Theory of Human Stereo Vision," *Proc. Royal Soc. London*, Vol. B204, pp. 301-328, 1979.

[11] McClelland, J. and D. Rumelhart, "An Interactive Activation Model of Context Effects in Letter Perception: Part 1. An Account of Basic Findings," *Psychological Review*, 1988.

[12] Hinton, G. and T. Sejnowski, "Learning and Relearning in Boltzmann Machines," D. Rumelhart and J. McClelland, in *Parallel Distributed Processing*, Vol. 1, Ch 7., MIT Press, 1986.

[13] Hopfield, J. and D. Tank, "Neural Computation of Decisions in Optimization Problems," *Biological Cybernetics*, Vol. 52, 1985, pp. 141-152.

[14] Anderson, J., "Neural Models with Cognitive Implications," in D. LaBerge and S. Samuels (Eds.), *Basic Processes in Reading Perception and Comprehension*, pp. 27-90, Hillsdale, NJ: Erlbaum, 1977.

[15] Grossberg, S., "A Theory of Visual Coding, Memory, and Development," E. Leeuwenberg and H. Buffart (Eds.), *Formal Theories of Visual Perception*, New York: Wiley, 1978.

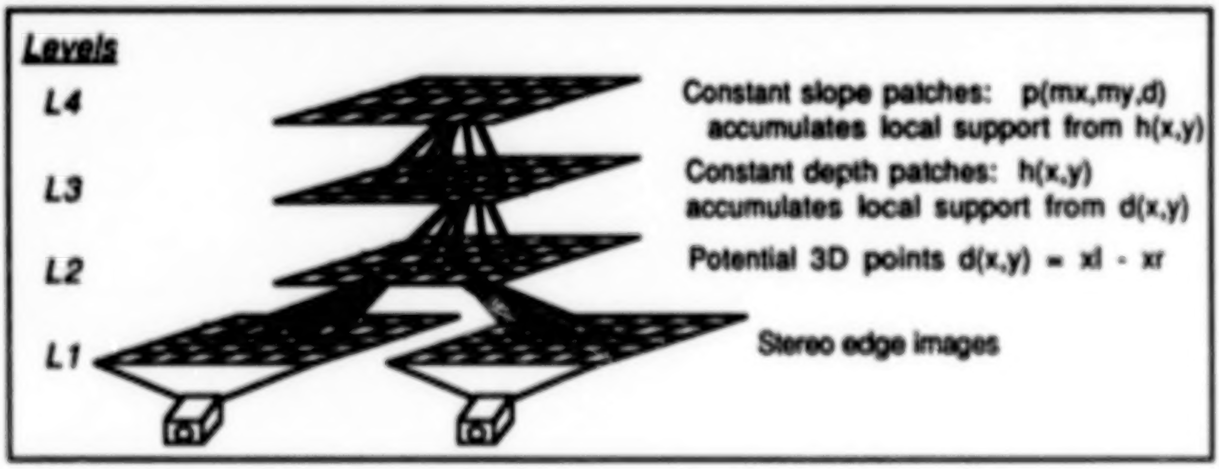

Levels
L4
L3
L2
L1
Constant slope patches: p(mx,my,d) accumulates local support from h(x,y)
Constant depth patches: h(x,y) accumulates local support from d(x,y)
Potential 3D points d(x,y) = xl - xr
Stereo edge images

Figure 1. Four level hierarchical model for stereo matching and surface representation.

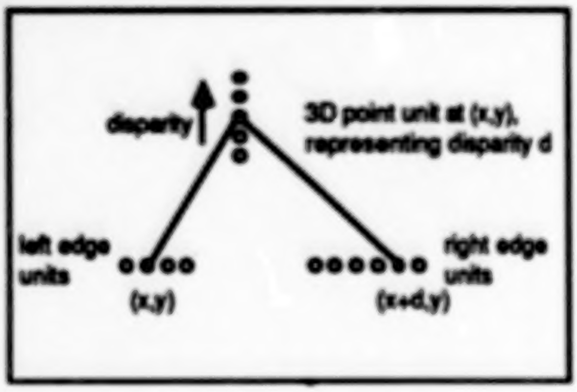

disparity
3D point unit at (x,y), representing disparity d
left edge units
(x,y)
right edge units
(x+d,y)

Figure 2. A 3D point unit receives support from left and right edge point units.

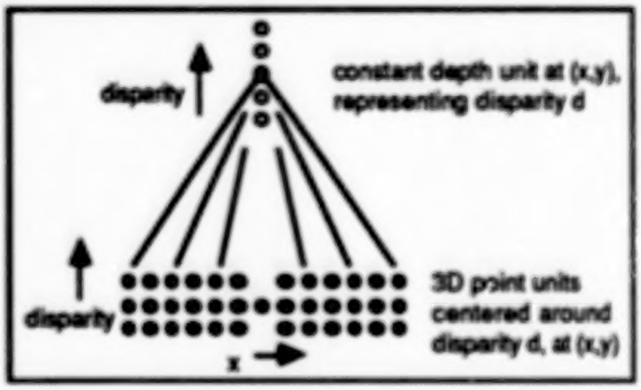

disparity
constant depth unit at (x,y), representing disparity d
disparity
3D point units centered around disparity d, at (x,y)
x

Figure 3. A constant-depth patch unit receives support from 3D point units.

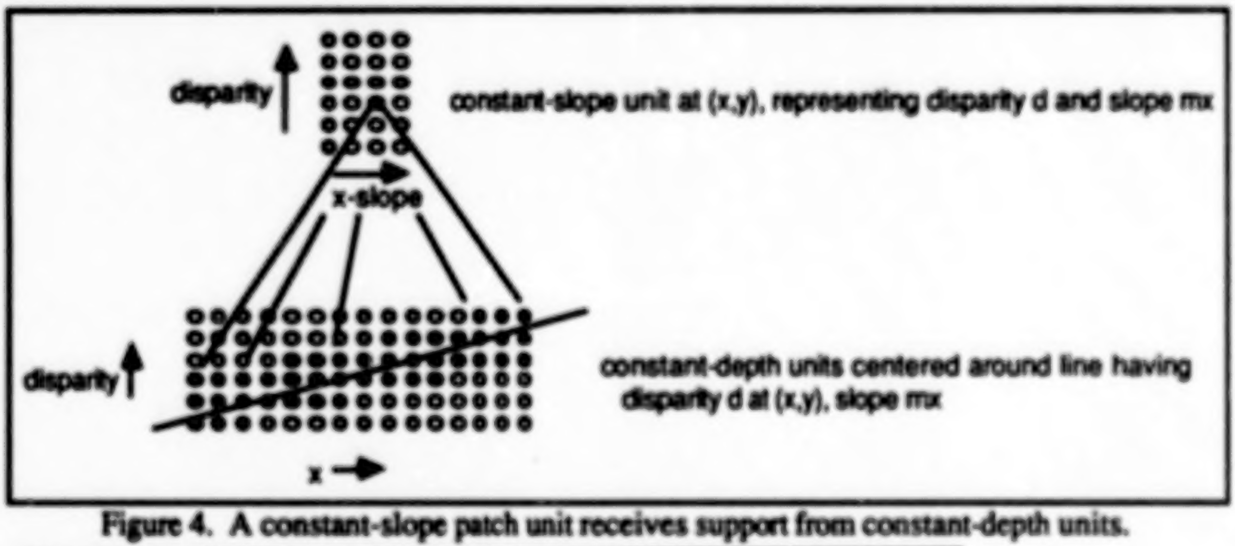

disparity
constant-slope unit at (x,y), representing disparity d and slope mx
x-slope
disparity
constant-depth units centered around line having disparity d at (x,y), slope mx
x

Figure 4. A constant-slope patch unit receives support from constant-depth units.

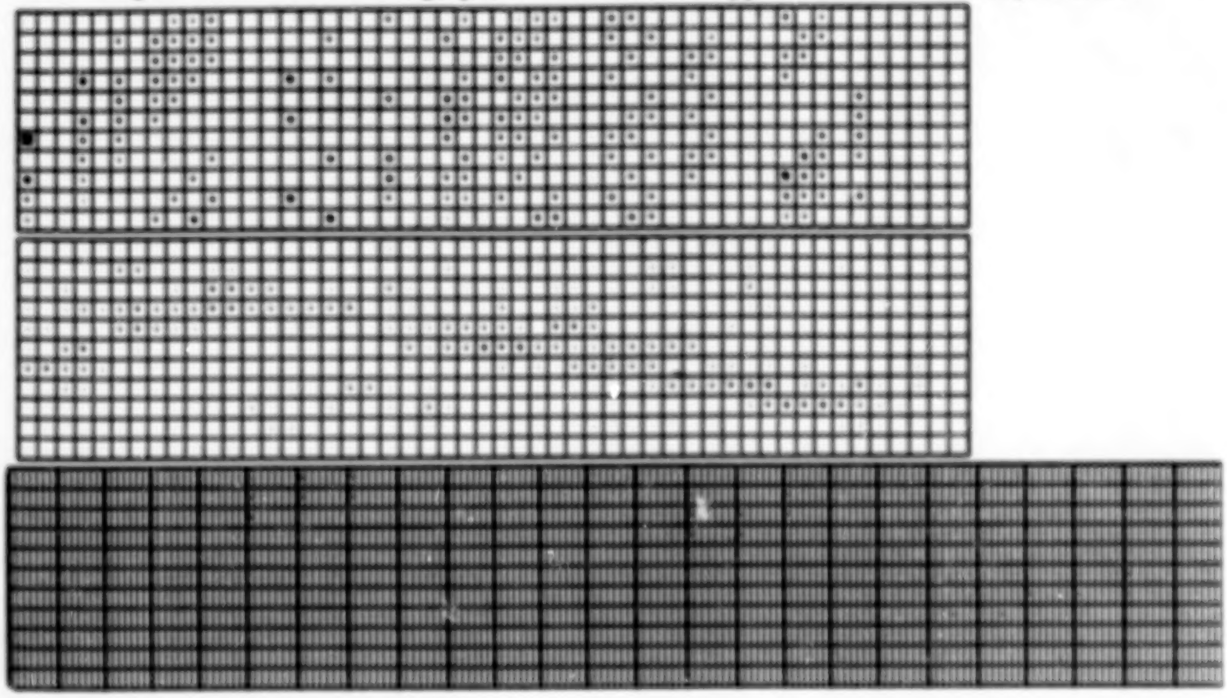


Figure 5a. The network after 1 iteration.

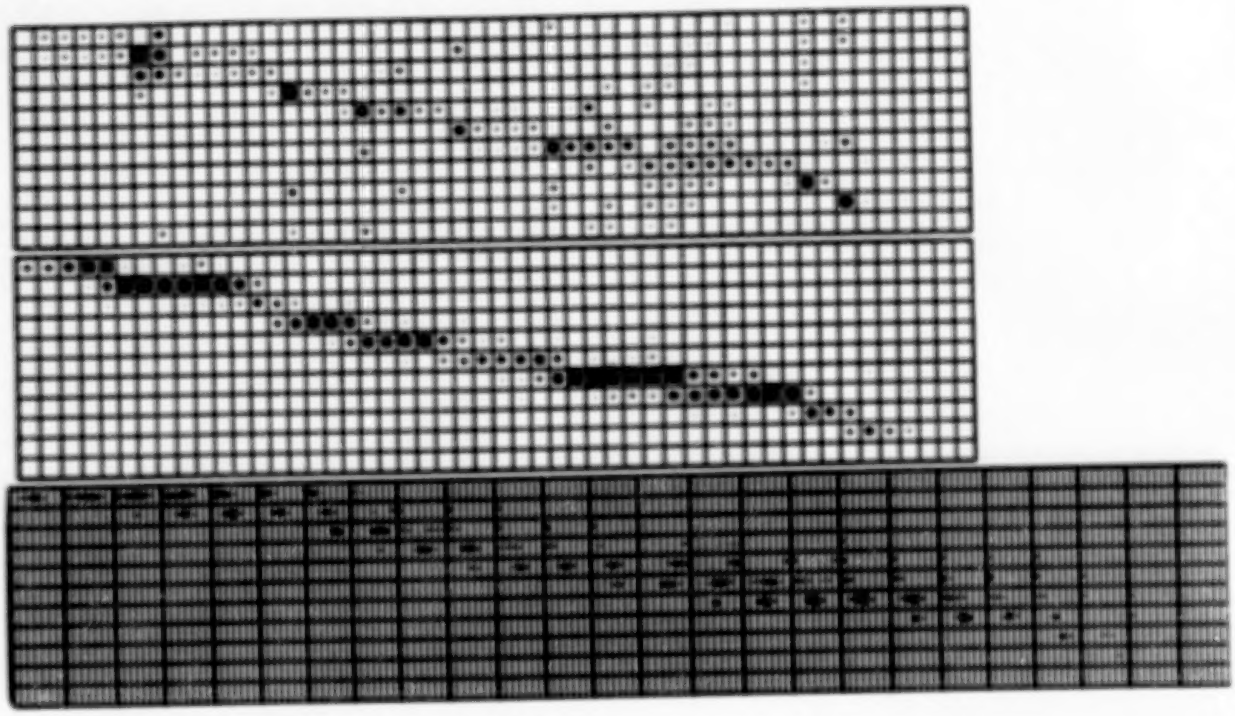


Figure 5b. The network after 5 iterations.

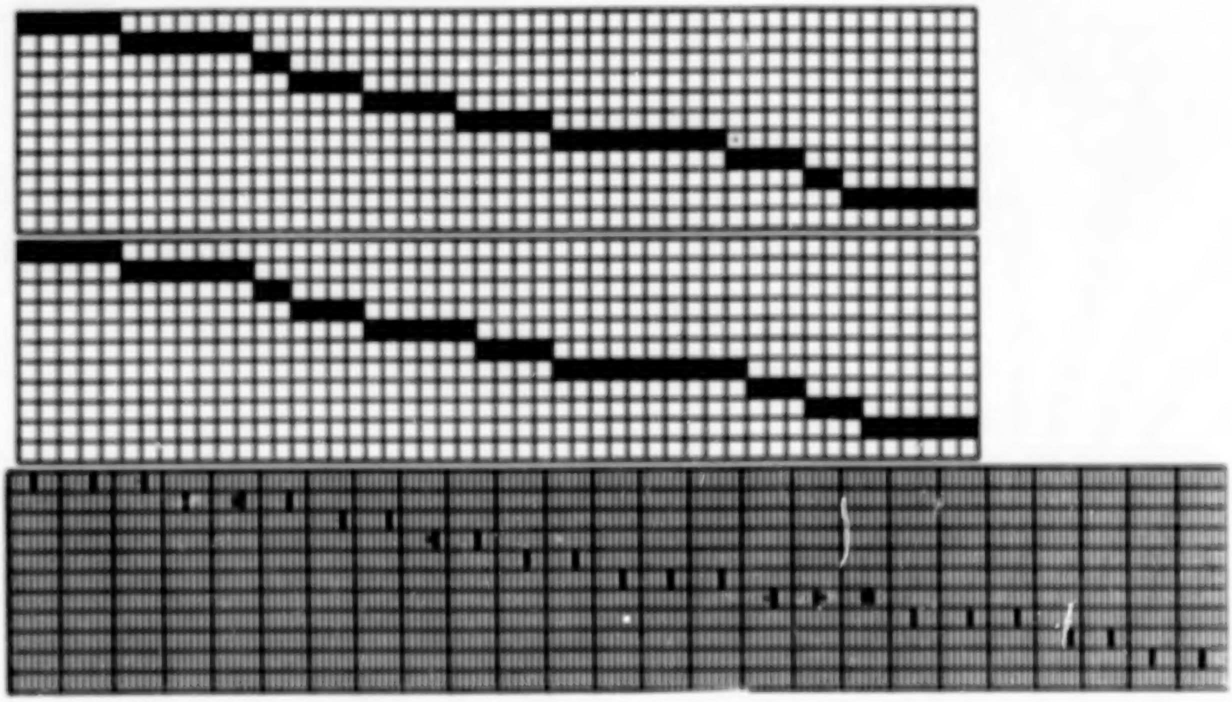


Figure 5c. The network after 9 iterations.

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.
NASA CP-3053

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
Visual Information Processing for Television
and Telerobotics

5. Report Date
November 1989

6. Performing Organization Code

7. Author(s)
Friedrich O. Huck and Stephen K. Park, Editors

8. Performing Organization Report No.
L-16665

10. Work Unit No.
549-02-21-01
694-40-01-01

9. Performing Organization Name and Address
NASA Langley Research Center
Hampton, VA 23665-5225

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546-0001

13. Type of Report and Period Covered
Conference Publication

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

This publication is a compilation of the papers presented at the NASA conference on Visual Information Processing for Television and Telerobotics. The conference was held at the Williamsburg Hilton, Williamsburg, Virginia, on May 10-12, 1989. The conference was sponsored jointly by NASA Offices of Aeronautics and Space Technology (OAST) and Space Science and Applications (OSSA) and the NASA Langley Research Center. The presentations were grouped into three sessions: Image Gathering, Coding, and Processing; Vision Models and Image Coding; and Advanced Concepts, Systems, and Technologies. The program was organized to provide a forum in which researchers from industry, universities, and government could be brought together to discuss the state of knowlege in image gathering, coding, and processing methods. Emphasis was on applications to high-resolution television and vision-based telerobotics and on the identification of areas in which further research is needed.

17. Key Words (Suggested by Author(s))
Image gathering, image coding,
image restoration

18. Distribution Statement
Unclassified-Unlimited
Subject Category 35

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	275	A12

NASA FORM 1626 OCT 86 For sale by the National Technical Information Service, Springfield, Virginia 22161-2171 NASA-Langley, 1989